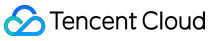
Key Management Service

TCCLI Management Guide

Product Documentation

Copyright Notice

Trademark Notice



Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

TCCLI Management Guide
Operation Overview

Last updated：2024-01-11 16:28:54

You can call KMS TCCLI to manage your keys, such as creating/editing/rotating a key and viewing the key ID list.
The operations below are called with TCCLI which can also be called with any supported programming languages.

Operation	Description
Creating key	Describes how to call TCCLI to create a key
Viewing key	Describes how to call TCCLI to view the key ID and details
Editing key	Describes how to call TCCLI to edit a key
Enabling/disabling key	Describes how to call TCCLI to enable/disable a key
Rotating key	Describes how to call TCCLI to rotate a key
Encryption and decryption	Describes how to call TCCLI for encryption and decryption
Deleting key	Describes how to call TCCLI to delete a key

Creating Key

Last updated：2024-01-11 16:28:54

Overview

The CreateKey API can be called to create a customer master key (CMK) used for DEK management. The CMK can be used in other APIs to create DEKs, perform encryption and decryption, and do more.
The `Alias` parameter is required for this API. You can add other descriptions for the CMK as instructed in the CreateKey API document.
The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

This example shows you how to create a key named `test-gz01` in Guangzhou region with the description `this is test for gz key`.

Input

```
tccli kms CreateKey --region ap-guangzhou --Alias test-gz01 --
Description 'this is test for gz key'
```

Output

After creation, the key will be enabled by default, with the key rotation feature disabled.

```
{
    "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09",
    "Description": "this is test for gz key",
    "Alias": "test-gz01",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "RequestId": "994bbd90-7c8e-4522-85f2-c712da23f863",
    "KeyState": "Enabled",
    "CreateTime": 1571903621
}
```

Viewing Key

Last updated：2024-01-11 16:28:54

Overview

API Name	Description	Note
ListKeys	Shows the list of keys (KeyId information) under an account.	There are no required parameters for this API. For more information, please see the ListKeys API document.
DescribeKey	Views the details of the specified CMK, including CMK name, ID, status, and region.	The `KeyId` parameter is required for this API. For more information, please see the DescribeKey API document.

The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Viewing the list of key IDs

This example describes how to view the information of the first five KeyIds in Guangzhou region.

Input

```
tccli kms ListKeys --region ap-guangzhou --Limit 5
```

Output

```
{
    "Keys": [
        {
            "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09"
        },
        {
            "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09"
        },
        {
            "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09"
```

```
        },
        {
            "KeyId":"6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09"
        },
        {
            "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09"
        }
    ],
    "TotalCount": 114,
    "RequestId": "afaaeb5e-c97d-4726-8012-6ae337d62928"
}
```

Viewing key ID details

This example describes how to view the details of the specified CMK.

Input

```
tccli kms DescribeKey --region ap-guangzhou --KeyId 521xxxxx-xxxx-xxxx-
xxxx-52xxxxd4
```

Output

If the API is successfully executed, the details of the CMK will be returned.

```
{
 "KeyMetadata": {
        "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09",
        "Description": "this is test for gz key",
        "CreatorUin": 10xxxxxxxxxx,
        "KeyRotationEnabled": false,
        "NextRotateTime": 1603439621,
        "CreateTime": 1571903621,
        "Alias": "test-gz01",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "DeletionDate": 0,
        "KeyState": "Enabled",
        "Type": 4,
        "Owner": "user"
    },
```

```
    "RequestId": "608f514c-3279-44ea-8e4c-c00b69e3521c"
}
```

Editing Key

Last updated：2024-01-11 16:28:54

Overview

The operations of renaming a key and modifying key description involve the following two functions:

API Name	Description	Note
UpdateAlias	Renames a key	The `KeyId` and `Alias` parameters are required for this API. For more information, please see the UpdateAlias API document.
UpdateKeyDescription	Modifies key description	The `KeyId` and `Description` parameters are required for this API. For more information, please see the UpdateKeyDescription API document.

The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Renaming a key

Input

```
tccli kms UpdateAlias --region ap-guangzhou --KeyId 52xxxx-xxxx-xxxx-
xxxx-5xxxx4 --Alias test-gz-01-d
```

Output

If the modification is successful, the following information will be returned.

```
{
    "RequestId": "489a4274-0b81-4db7-8160-542c5c5bed68"
}
```

Modifying key description

Input

```
tccli kms UpdateKeyDescription --region ap-guangzhou --KeyId 5xxxxx-
xxxx-xxxx-xxxx-52xxxxx4 --Description 'this is change message for test'
```

Output

If the modification is successful, the following information will be returned.

```
{
    "RequestId": "31134207-5de8-44f2-8c00-8bd0e88f95a6"
}
```

Enabling/Disabling Key

Last updated：2024-01-11 16:28:54

Overview

The operations of enabling and disabling a key involve the following two APIs:

API Name	Description	Note
EnableKey	Enables a CMK	The `KeyId` parameter is required for this API. For more information, please see the EnableKey API document.
DisableKey	Disables a CMK	The `KeyId` parameter is required for this API. For more information, please see the DisableKey API document.

The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Enabling a CMK

Input

```
tccli kms EnableKey --region ap-guangzhou --KeyId 5xxxxx-xxxx-xxxx-xxxx-
52xxxxx4
```

Output

If the key is successfully enabled, the following request will be returned.

```
{
    "RequestId": "6b2187b0-f40a-46d0-8065-2434afc54619"
}
```

Disabling a CMK

Input

```
tccli kms DisableKey --region ap-guangzhou --KeyId 5xxxxx-xxxx-xxxx-
xxxx-52xxxxx4
```

Output

If the key is successfully disabled, the following request will be returned.

```
{
    "RequestId": "e5674638-1466-4607-a3ea-b60d30f4e5e3"
}
```

Key Rotation

Last updated：2024-01-11 16:28:54

Overview

The key rotation feature involves three APIs:

API Name	Description	Note
GetKeyRotationStatus	Views key rotation status	The `KeyId` parameter is required for this API. For more information, please see the GetKeyRotationStatus API document.
EnableKeyRotation	Enables key rotation	The `KeyId` parameter is required for this API. For more information, please see the EnableKeyRotation API document.
DisableKeyRotation	Disables key rotation	The `KeyId` parameter is required for this API. For more information, please see the DisableKeyRotation API document.

The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Viewing key rotation status

Input

```
tccli kms GetKeyRotationStatus --region ap-guangzhou --KeyId 5xxxxx-xxxx-xxxx-xxxx-52xxxxx4
```

Output

If the API is called successfully, the key rotation status of the CMK will be returned.

```
{
    "KeyRotationEnabled": false,
    "RequestId": "e1432224-4dc2-48da-a8e8-e84d30afd9ef"
}
```

Enabling key rotation

Input

```
tccli kms EnableKeyRotation --region ap-guangzhou --KeyId 5xxxxx-xxxx-
xxxx-xxxx-52xxxxx4
```

Output

If the feature is enabled normally, the request information as shown below will be returned.

```
{
    "RequestId": "4e0fa96f-e86e-4517-af27-3dfe6e5b2a72"
}
```

Disabling key rotation

Input

```
tccli kms DisableKeyRotation --region ap-guangzhou --KeyId 5xxxxx-xxxx-
xxxx-xxxx-52xxxxx4
```

Output

If the feature is disabled normally, the request information as shown below will be returned.

```
{
    "RequestId": "c8b73c8b-1ee5-4b23-b800-7cccc58e7ffb"
}
```

Encryption and Decryption

Last updated：2024-01-11 16:28:54

Overview

The online encryption and decryption operations involve two APIs:

API Name	Description	Note
Encrypt	Used for encryption	The `KeyId` and `Plaintext` parameters are required for this API. For more information, please see the Encrypt API document.
Decrypt	Used for decryption	The `CiphertextBlob` parameter is required for this API. For more information, please see the Decrypt API document.

Encryption

The Encrypt API is used to encrypt up to 4 KB of data, such as database passwords, RSA keys, or other sensitive data. For application data, the DEK generated by the GenerateDataKey API can be used to perform encryption and decryption for the local data.
The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Encryption

If the Encrypt API is called with TCCLI, the plaintext data needs to be Base64-encoded. The `This example is used for testing` text is used in the following example.

Input

```
tccli kms Encrypt --KeyId 6xxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxx5 --Plaintext
'VGhpcyBleGFtcGxlIGlzIHVzZWQgZm9yIHRlc3Rpbmc='
```

Output

If the execution is successful, the ciphertext and the CMK ID used to encrypt the plaintext will be returned, of which the ciphertext will be used for subsequent decryption operations.

```
{
```

```
    "KeyId": "6xxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxx5",
    "RequestId": "23781471-c213-44c5-92a4-731b882e25b5",
    "CiphertextBlob":
"Rrnqz5fthTxcSdCYIw5pBoEWLvrdqYNZ0oXKOmvYx/1Oo2R+DqEFPjjfVA1n1RE8PmVITax
uJwu9ZANK9uK3WA==-k-fKVP3WIlGpg8m9LMW4jEkQ==-k-
mFM/5PEiMJsKC6fagEOfdlocOyC+a1n8PqaTOlBLT+rqjyKLVHUVtqamMQ3ERsYIe0wYoAMs
zR/FBrCJZ3a3B7f+8Xg="
}
```

Decryption

This example shows you how to decrypt the encrypted data, where the CMK is the one used in the above example.

Input

```
tccli kms Decrypt --CiphertextBlob
'Rrnqz5fthTxcSdCYIw5pBoEWLvrdqYNZ0oXKOmvYx/1Oo2R+DqEFPjjfVA1n1RE8PmVITax
uJwu9ZANK9uK3WA==-k-fKVP3WIlGpg8m9LMW4jEkQ==-k-
mFM/5PEiMJsKC6fagEOfdlocOyC+a1n8PqaTOlBLT+rqjyKLVHUVtqamMQ3ERsYIe0wYoAMs
zR/FBrCJZ3a3B7f+8Xg='
```

Output

If the execution is successful, the Base64-encoded plaintext and the CMK ID used to encrypt the plaintext will be returned. An additional decryption operation in Base64 is needed to obtain the plaintext.

```
{
    "Plaintext": "VGhpcyBleGFtcGxlIGlzIHVzZWQgZm9yIHRlc3Rpbmc=",
    "KeyId": "6xxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxx5",
    "RequestId": "bcce3fae-1794-4136-a486-d42780c10702"
}
```

Asymmetric key decryption

Last updated：2024-01-11 16:28:54

Overview

KMS provides the following SM2 and RSA asymmetric key-based decryption APIs:

API Name	Description	Remarks
AsymmetricSm2Decrypt	SM2 decryption	For more information, please see AsymmetricSm2Decrypt
AsymmetricRsaDecrypt	RSA decryption	For more information, please see AsymmetricRsaDecrypt

The samples below are called with TCCLI, and you can also use any supported programming languages.

Asymmetric Decryption

RSA decryption

Input

```
tccli kms AsymmetricRsaDecrypt --KeyId 22d79428-61d9-11ea-a3c8-
525400****** --Algorithm RSAES_OAEP_SHA_256 --Ciphertext
"DEb/JBmuhVkYS34r0pR7Gv1WTc4khkxqf7S1WIr7/GXsAs/tfP/v/2+1SwsIG7BqW7kUZqr
38/FGkaIEqYeewot37t3+Jx0t5w7/yXkUnyUfyfPpXlHXf94g3wFOjijEWWsjWWzaXTkTr8u
WOfRBenq+bcaY783FIy03XjJW/Y0wKWjD3tULvKndCJO/3bkb65kn1Fbsfm20xrUUwqV/p2D
VLXBdG1ymr0DjsbG7R0tb3ytc2LmH33YPAQE32eP27ciKzSml+w2tdUM3dw3nEZcTGMs1wFD
Gk0O1WB052jZ7TitUD9zCftFv2dKlZD3LRx1+vHqpNVgPhLmL******=="
```

Output

```
{
    "Response": {
        "RequestId": "6758cbf5-5e21-4c37-a2cf-8d47f5******",
        "KeyId": "22d79428-61d9-11ea-a3c8-525400******",
        "Plaintext": "dGVzdAo="
    }
}
```

SM2 decryption

Input

```
tccli kms AsymmetricSm2Decrypt --KeyId 22d79428-61d9-11ea-a3c8-
525400****** --Ciphertext
"DEb/JBmuhVkYS34r0pR7Gv1WTc4khkxqf7S1WIr7/GXsAs/tfP/v/2+1SwsIG7BqW7kUZqr
38/FGkaIEqYeewot37t3+Jx0t5w7/yXkUnyUfyfPpXlHXf94g3wFOjijEWWsjWWzaXTkTr8u
WOfRBenq+bcaY783FIy03XjJW/Y0wKWjD3tULvKndCJO/3bkb65kn1Fbsfm20xrUUwqV/p2D
VLXBdG1ymr0DjsbG7R0tb3ytc2LmH33YPAQE32eP27ciKzSml+w2tdUM3dw3nEZcTGMs1wFD
Gk0O1WB052jZ7TitUD9zCftFv2dKlZD3LRx1+vHqpNVgPhLmL******=="
```

Output

```
{
    "Response": {
        "RequestId": "6758cbf5-5e21-4c37-a2cf-8d47f5******",
        "KeyId": "22d79428-61d9-11ea-a3c8-525400******",
        "Plaintext": "dGVzdAo="
    }
}
```

Viewing Public Key

Overview

This API is used to get the information of the public key with the specified `KeyId`. For the API documentation, please see GetPublicKey.
The sample below is called with TCCLI, and you can also use any supported programming languages.

Sample

Input

```
tccli kms GetPublicKey  --KeyId 22d79428-61d9-11ea-a3c8-525400******
```

Output

```
{
    "Response": {
```

```
        "RequestId": "408fa858-cd6d-4011-b8a0-653805******",
        "KeyId": "22d79428-61d9-11ea-a3c8-525400******",
        "PublicKey":
"MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzQk7x7ladgVFEEGYDbeUc5aO9Tf
iDplIO4WovBOVpIFoDS31n46YiCGiqj67qmYslZ2KMGCd3Nt+a+jdzwFiTx3O87wdKWcF2vH
L9Ja+95VuCmKYeK1uhPyqqj4t9Ch/cyvxb0xaLBzztTQ9dXCxDhwj08b24T+/FYB9a4icuqQ
ypCvjY1X9j8ivAsPEdHZoc9Di7JXBTZdVeZC1igCVgl6mwzdHTJCRydE2976zyjC7l6QsRT6
pRsMF3696N07WnaKgGv3K/Zr/6RbxebLqtmNypNERIR7jTCt9L+fgYOX7anmuF5v7z0GfFse
n9Tqb1LsZuQR0vgqCauOjL2CL1Q******",
        "PublicKeyPem": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzQk7x7ladgVFEEGYDbeU\nc5aO
9TfiDplIO4WovBOVpIFoDS31n46YiCGiqj67qmYslZ2KMGCd3Nt+a+jdzwFi\nTx3O87wdKW
cF2vHL9Ja+95VuCmKYeK1uhPyqqj4t9Ch/cyvxb0xaLBzztTQ9dXCx\nDhwj08b24T+/FYB9
a4icuqQypCvjY1X9j8ivAsPEdHZoc9Di7JXBTZdVeZC1igCV\ngl6mwzdHTJCRydE2976zyj
C7l6QsRT6pRsMF3696N07WnaKgGv3K/Zr/6RbxebLq\ntmNypNERIR7jTCt9L+fgYOX7anmu
F5v7z0GfFsen9Tqb1LsZuQR0vgqCau******\n1QIDAQAB\n-----END PUBLIC KEY-----
\n"
    }
}
```

Deleting Key

Last updated：2024-01-11 16:28:54

Overview

The schedule key deletion feature involves the following two APIs:

API Name	Description	Note
ScheduleKeyDeletion	Creates a schedule deletion task	The `KeyId` and `PendingWindowInDays` parameters are required for this API.
CancelKeyDeletion	Cancels a schedule deletion task	The `KeyId` parameter is required for this API.

Note：
If a CMK schedule deletion waiting period is set through the ScheduleKeyDeletion API when the CMK is in disabled status, the CMK will be deleted automatically at the specified time.

The examples below are called with TCCLI, which can also be called with any supported programming languages.

Examples

Creating a schedule deletion task

This example shows you how to delete a disabled CMK in 7 days.

Input

```
tccli kms ScheduleKeyDeletion --region ap-guangzhou --KeyId 5xxxxx-xxxx-xxxx-xxxx-52xxxxx4 --PendingWindowInDays 7
```

Output

If the setting is successful, the ID of the CMK to be deleted and the schedule deletion timestamp will be returned.

```
{
    "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09",
```

```
    "RequestId": "2bd72d85-f9dd-4465-ae51-beebff54f540",
    "DeletionDate": 1572512542
}
```

Canceling a schedule deletion task

This example shows you how to cancel a schedule deletion task, where the CMK is the one used in the above example.

Input

```
tccli kms CancelKeyDeletion --region ap-guangzhou --KeyId 5xxxxx-xxxx-
xxxx-xxxx-52xxxxx
```

Output

If the execution is successful, the returned request will contain the ID of the CMK for which the schedule deletion task is successfully canceled.

```
{
    "KeyId": "6xxxxxxx-xxxx-xxxx-xxxx-5xxxxxxxxc09",
    "RequestId": "c85473c6-e18d-4a09-9eac-03958dd4714d"
}
```