TDMQ for CMQ

SDK Documentation

Product Documentation

Contents

SDK Documentation

HTTP Data Flow SDK

API Overview

Last updated：2024-01-03 10:20:36

TDMQ for CMQ currently supports SDKs for Java, Python, PHP, and C++. More languages will be supported in the future. Developers are encouraged to develop SDKs for more languages based on the API descriptions.
To allow CMQ users to migrate to TDMQ for CMQ without modifying the business code, the following APIs remain the same as those on earlier versions.

API Feature	Action ID	Description
Sending message	SendMessage	Sends a message to a specified queue.
Batch sending messages	BatchSendMessage	Batch sends messages to a specified queue.
Consuming message	ReceiveMessage	Consumes a message in a queue.
Batch consuming messages	BatchReceiveMessage	Batch consumes messages in a queue.
Deleting message	DeleteMessage	Deletes a consumed message.
Batch deleting messages	BatchDeleteMessage	Batch deletes consumed messages.
Publishing message	PublishMessage	Publishes a message to a specified topic.
Batch publishing messages	BatchPublishMessage	Batch publishes messages to a specified topic.

Other APIs need to be developed as instructed in HTTP Control Flow SDK.

SDK for Java

Last updated：2024-01-03 10:20:36

Overview

This document uses the SDK for Java as an example to describe how to connect the client to TDMQ for CMQ and send/receive messages.

Prerequisites

You have installed JDK 1.8 or later.
You have installed Maven 2.5 or later.
You have downloaded the demo.

Queue Model

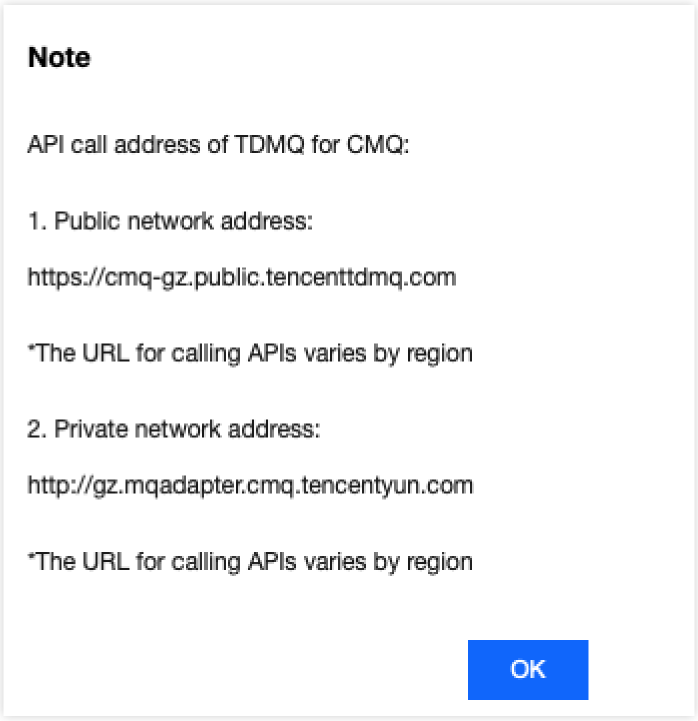
Directions

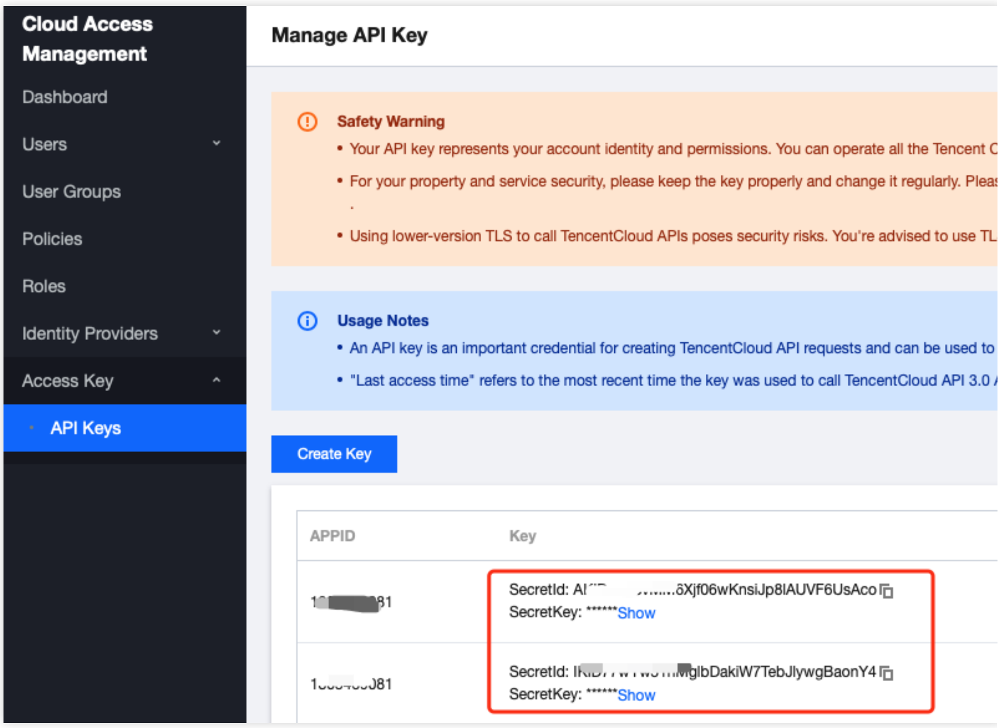
1. Create a queue service in the console as instructed in Queue Management.
2. Import CMQ client dependencies.

```
<!-- cmq sdk -->
<dependency>
    <groupId>com.qcloud</groupId>
    <artifactId>cmq-http-client</artifactId>
    <version>1.0.7</version>
</dependency>

<!-- TencentCloud API sdk -->
<dependency>
    <groupId>com.tencentcloudapi</groupId>
    <artifactId>tencentcloud-sdk-java</artifactId>
    <version>3.1.423</version>
</dependency>
```

3. Send messages.

```
    Account account = new Account(SERVER_ENDPOINT, SECRET_ID, SECRET_KEY);
    Queue queue = account.getQueue(queueName);
    String msg = "hello client, this is a message. Time:" + new Date();
    CmqResponse response = queue.send(msg);
```

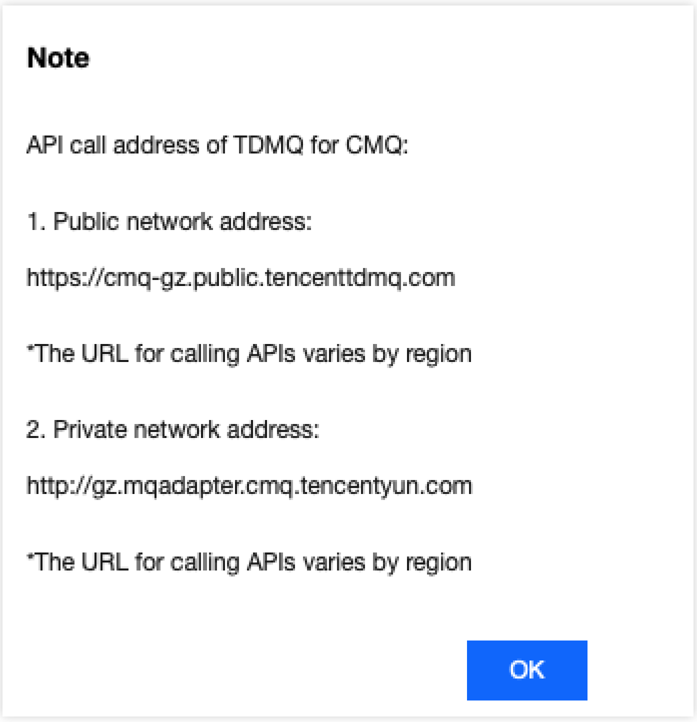
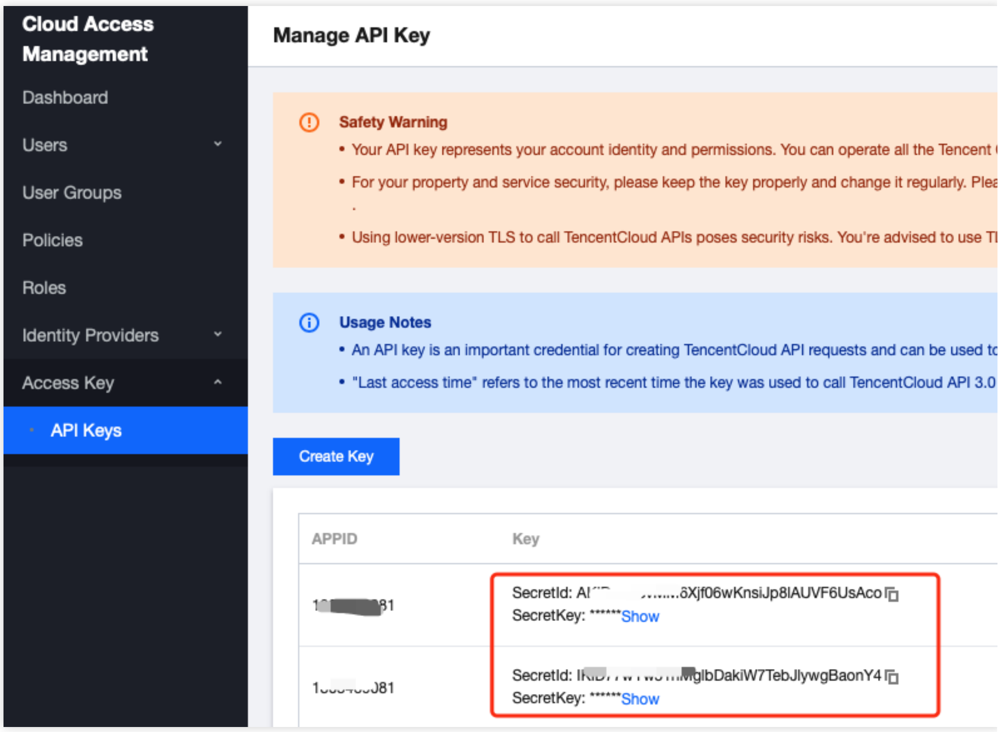
Parameter	Description
SERVER_ENDPOINT	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
SECRET_ID, SECRET_KEY	TencentCloud API key, which can be copied on the Access Key > API Key Management in the CAM console.

	 Cloud Access Management Dashboard Users User Groups Policies Roles Identity Providers Access Key API Keys Manage API Key Safety Warning • Your API key represents your account identity and permissions. You can operate all the Tencent C • For your property and service security, please keep the key properly and change it regularly. Pleas • Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TL Usage Notes • An API key is an important credential for creating TencentCloud API requests and can be used to • "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 Create Key APPID / Key 1[redacted]81 / SecretId: AI[redacted]8Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show 1[redacted]081 / SecretId: IK[redacted]MglbDakiW7TebJlywgBaonY4 SecretKey: ******Show
queueName	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ console.

4. Consume messages.

```
    Account account = new Account(SERVER_ENDPOINT, SECRET_ID, SECRET_KEY);
    Queue queue = account.getQueue(queueName);
    Message message = queue.receiveMessage();
    // Successfully consumed messages are deleted. Retained messages can be delivere
    queue.deleteMessage(message.receiptHandle);
```

Parameter	Description
SERVER_ENDPOINT	API call address, which can be copied from Queue Service > API Request Address in the

	TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region OK
SECRET_ID, SECRET_KEY	TencentCloud API key, which can be copied on the Access Key > API Key Management the CAM console.  Cloud Access Management Dashboard Users User Groups Policies Roles Identity Providers Access Key API Keys **Manage API Key** **Safety Warning** • Your API key represents your account identity and permissions. You can operate all the Tencent • For your property and service security, please keep the key properly and change it regularly. Plea • Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use T **Usage Notes** • An API key is an important credential for creating TencentCloud API requests and can be used to • "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 Create Key APPID | Key [redacted] | SecretId: [redacted]8Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show [redacted] | SecretId: [redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

queueName	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ console.

Topic Model

Directions

1. Create resources in the console.

1.1 Create a topic in the console as instructed in Topic Management.

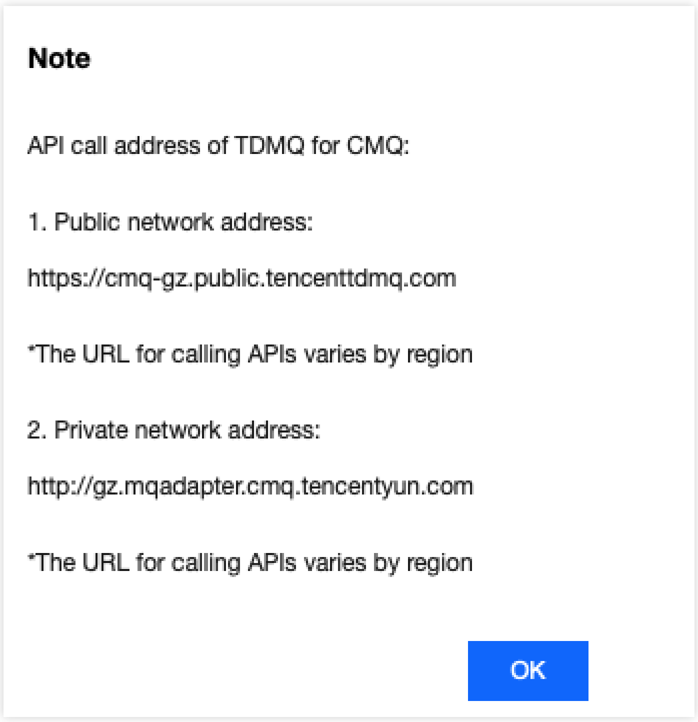
1.2 Create a subscriber for the topic as instructed in Subscription Management.

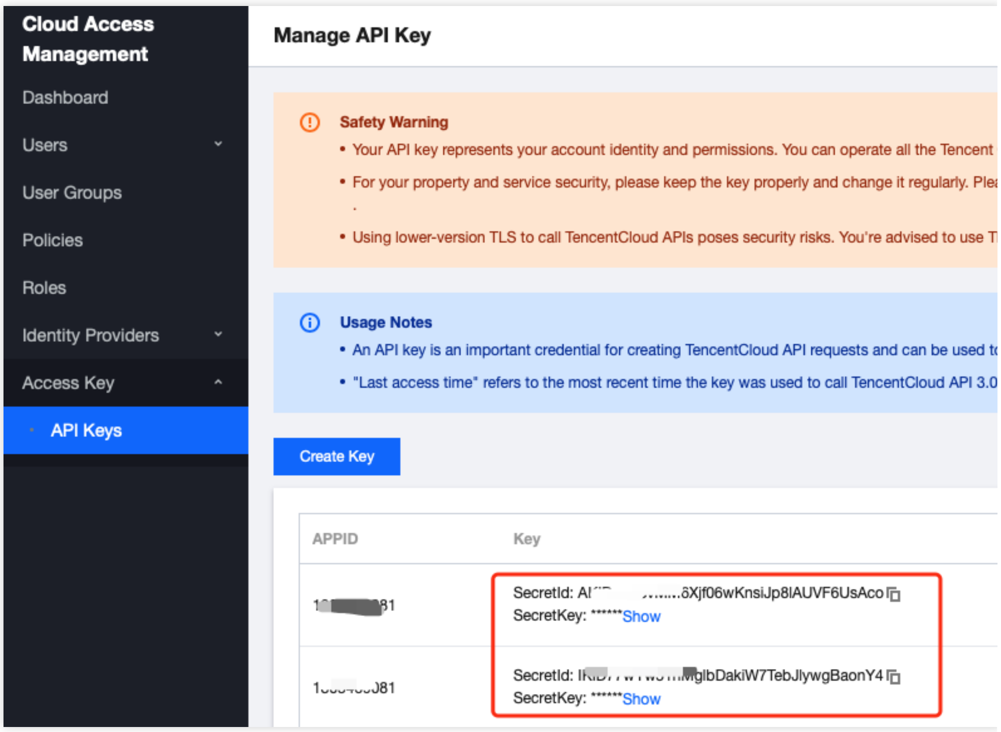
2. Import CMQ client dependencies.

```
<!-- cmq sdk -->
<dependency>
    <groupId>com.qcloud</groupId>
    <artifactId>cmq-http-client</artifactId>
    <version>1.0.7</version>
</dependency>

<!-- TencentCloud API sdk -->
<dependency>
    <groupId>com.tencentcloudapi</groupId>
    <artifactId>tencentcloud-sdk-java</artifactId>
    <version>3.1.423</version>
</dependency>
```

3. Create a topic object.

```
Account account = new Account(SERVER_ENDPOINT, SECRET_ID, SECRET_KEY);
Topic topic = account.getTopic(topicName);
```

Parameter	Description
SERVER_ENDPOINT	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
SECRET_ID, SECRET_KEY	TencentCloud API key, which can be copied on the Access Key > API Key Management the CAM console.

	
topicName	Topic subscription name, which can be obtained on the Topic Subscription page in the TI for CMQ console.

Cloud Access Management
Dashboard
Users
User Groups
Policies
Roles
Identity Providers
Access Key
API Keys

Manage API Key

Safety Warning
- Your API key represents your account identity and permissions. You can operate all the Tencent
- For your property and service security, please keep the key properly and change it regularly. Plea
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use T

Usage Notes
- An API key is an important credential for creating TencentCloud API requests and can be used t
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0

Create Key

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: IK[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

4. Send tag messages.

```
String msg = "hello client, this is a message. tag=TAG1. Time:" + new Date();
List<String> tags = Collections.singletonList("TAG1");
String messageId = topic.publishMessage(msg, tags, null);
```

5. Send route messages.

```
String msg = "hello client, this is a message. route(abc) Time:" + new Date();
String messageId = topic.publishMessage(msg, "abc");
```

6. Consume messages in the queue corresponding to the subscriber.

```
Account account = new Account(SERVER_ENDPOINT, SECRET_ID, SECRET_KEY);
Queue queue = account.getQueue(queueName);
Message message = queue.receiveMessage();
// Successfully consumed messages are deleted. Retained messages can be delivere
queue.deleteMessage(message.receiptHandle);
```

Note:

Above is a brief introduction to message production and consumption in two models. For more information, see Demo.

SDK for Python

Last updated：2024-01-03 10:20:35

Directions

This document uses the SDK for Python as an example to describe how to connect the client to TDMQ for CMQ and send/receive messages.

Prerequisites

You have installed Python.
You have installed pip.
You have downloaded the demo.

Queue Model

Directions

1. Create a queue in the console as instructed in Queue Management.

Note:

You can create a message queue in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK for Python.

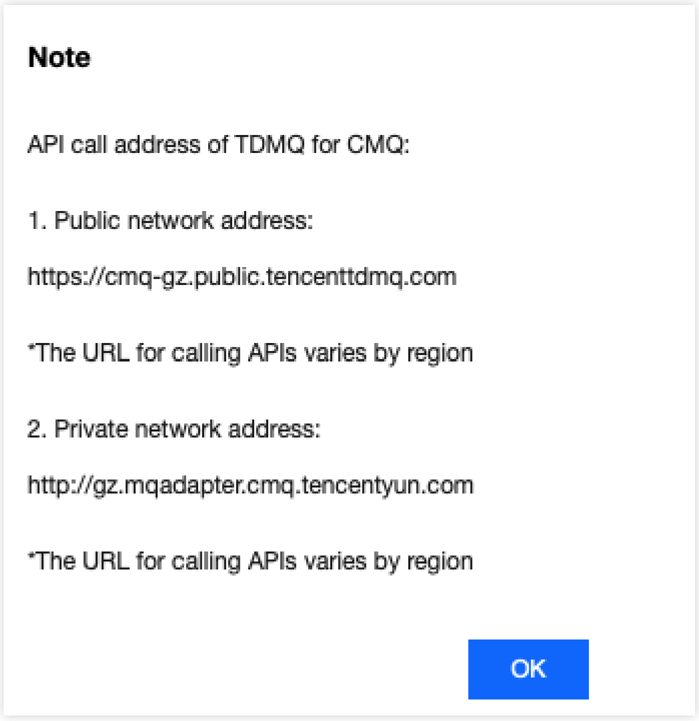
shell

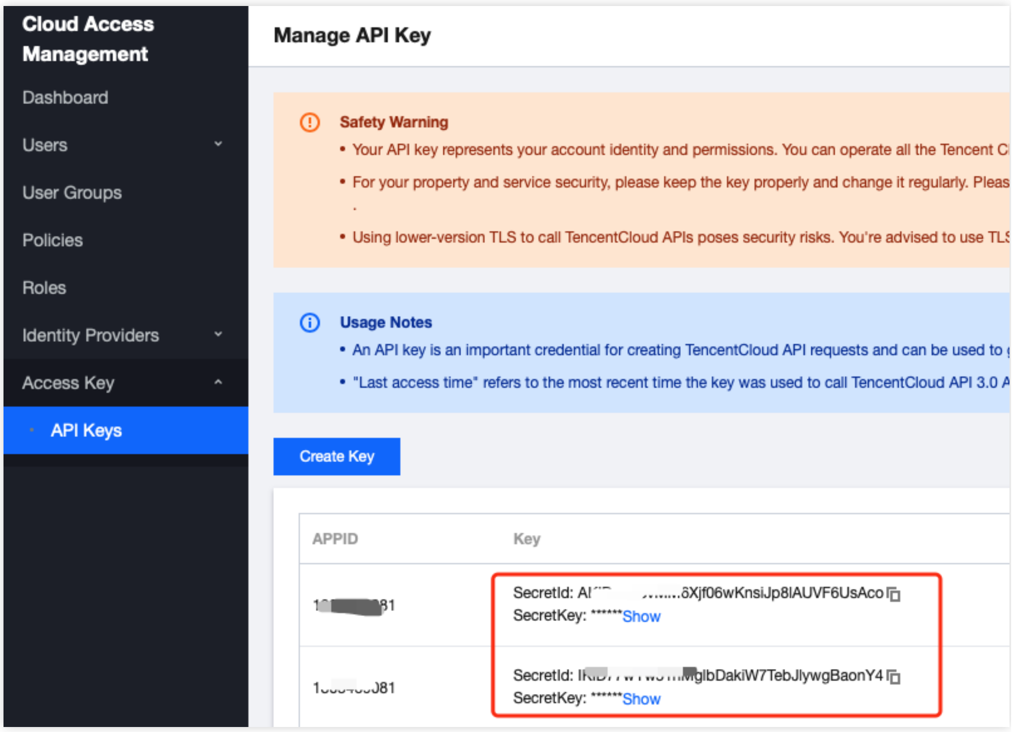
python

```
pip install --upgrade tencentcloud-sdk-python
```

```
# API authentication information
cred = credential.Credential(SecretId, SecretKey)
httpProfile = HttpProfile()
httpProfile.endpoint = NameServerAddress

clientProfile = ClientProfile()
clientProfile.httpProfile = httpProfile
# Create the TDMQ client
client = tdmq_client.TdmqClient(cred, "ap-guangzhou", clientProfile)

# Create CMQ queue request parameters
```

```
req = models.CreateCmqQueueRequest()
params = {
      "QueueName": "queue_api",
      # Below is the dead letter queue configuration
      "DeadLetterQueueName": "dead_queue_api",  # Dead letter queue, which needs to
      "Policy": 0,  # 0: Message has been consumed multiple times but not deleted. 1
      "MaxReceiveCount": 3  # Maximum receipts. Value range: 1-1000
}
req.from_json_string(json.dumps(params))

# Create a CMQ message queue
resp = client.CreateCmqQueue(req)
```

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.



Cloud Access Management

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

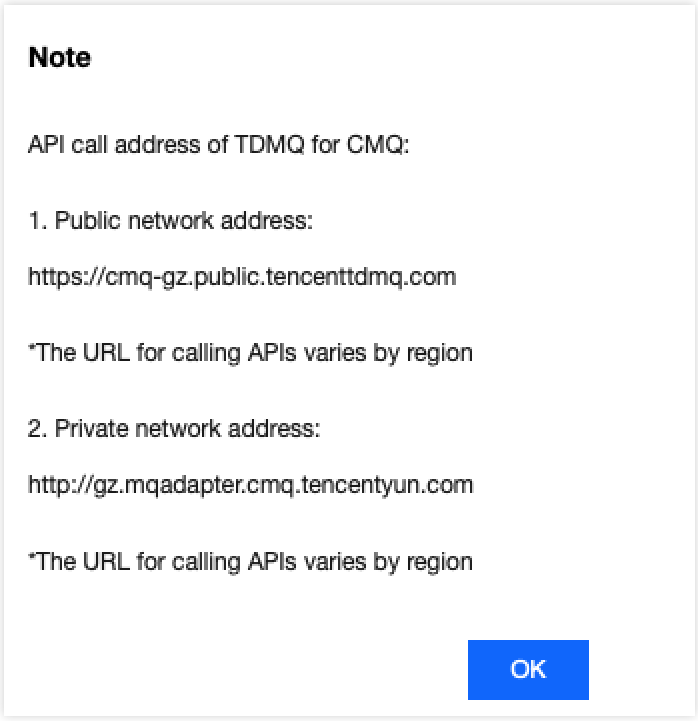
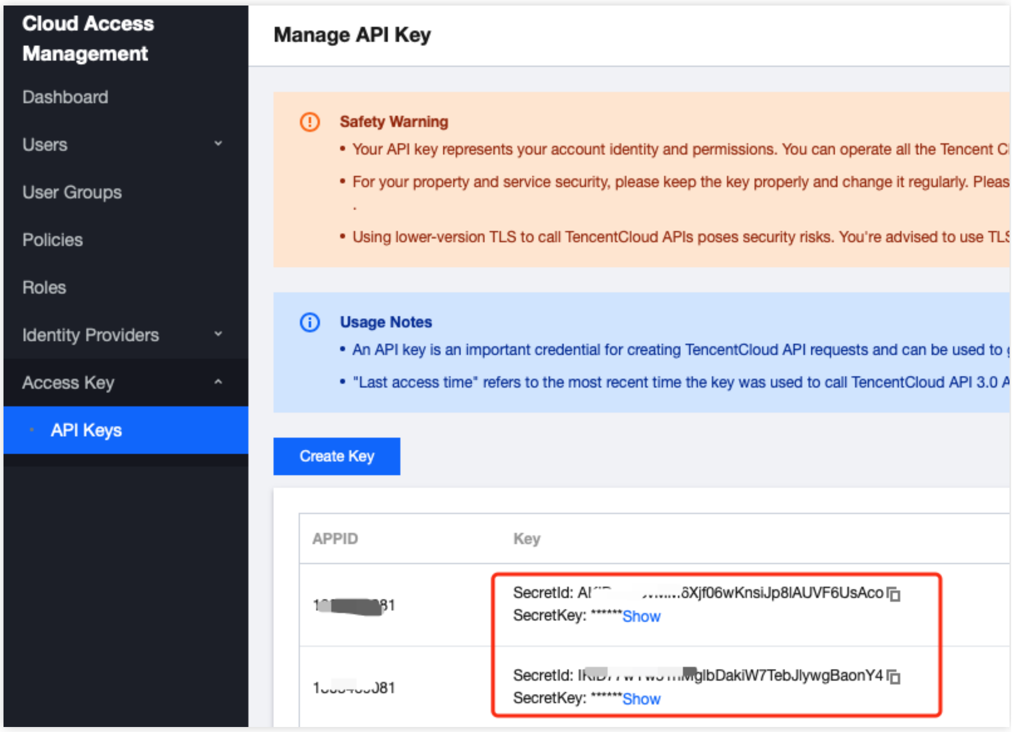
- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: I[redacted]MglbDakiW7TebJlywgBaonY4 SecretKey: ******Show

2. Import CMQ files into the project. You need to select a branch based on the used Python version. The default SDK is for Python 2, and you can switch to the Python 3 branch to view the SDK for Python 3.3. Send messages.

```
import os
import sys

sys.path.insert(0, os.path.dirname(os.path.abspath(__file__)) + "/..")

import logging
from cmq.account import Account
from cmq.queue import Message
from cmq.cmq_exception import CMQExceptionBase

# `secretId` and `secretKey` of your Tencent Cloud account, which should be kept co
# You can get them at https://console.intl.cloud.tencent.com/cam/capi
secretId = 'AKIDSiiRtxxxx'
secretKey = 'GGzSeaM5xxxx'
# CMQ service call address
nameServerAddress = 'https://cmq-gz.public.tencenttdmq.com'

# Initialize `my_account` and `my_queue`
# Account class objects are not thread-safe. If you need to use them in multiple th
my_account = Account(nameServerAddress, secretId, secretKey, debug=True)
my_account.set_log_level(logging.DEBUG)
# Message queue name
queue_name = sys.argv[1] if len(sys.argv) > 1 else "python_queue"
my_queue = my_account.get_queue(queue_name)

try:
     # Message content
     msg_body = "I am test message."
     msg = Message(msg_body)
     # Send messages
     re_msg = my_queue.send_message(msg)
     # Sending result
     print("Send Message Succeed! MessageBody:%s MessageID:%s" % (msg_body, re_msg.
except CMQExceptionBase as e:
     print("Send Message Fail! Exception:%s\\n" % e)
```

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.

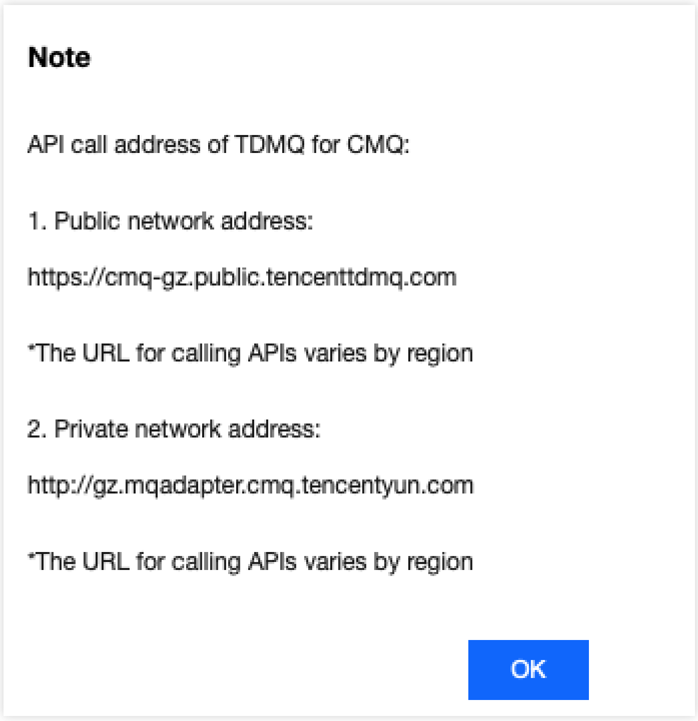
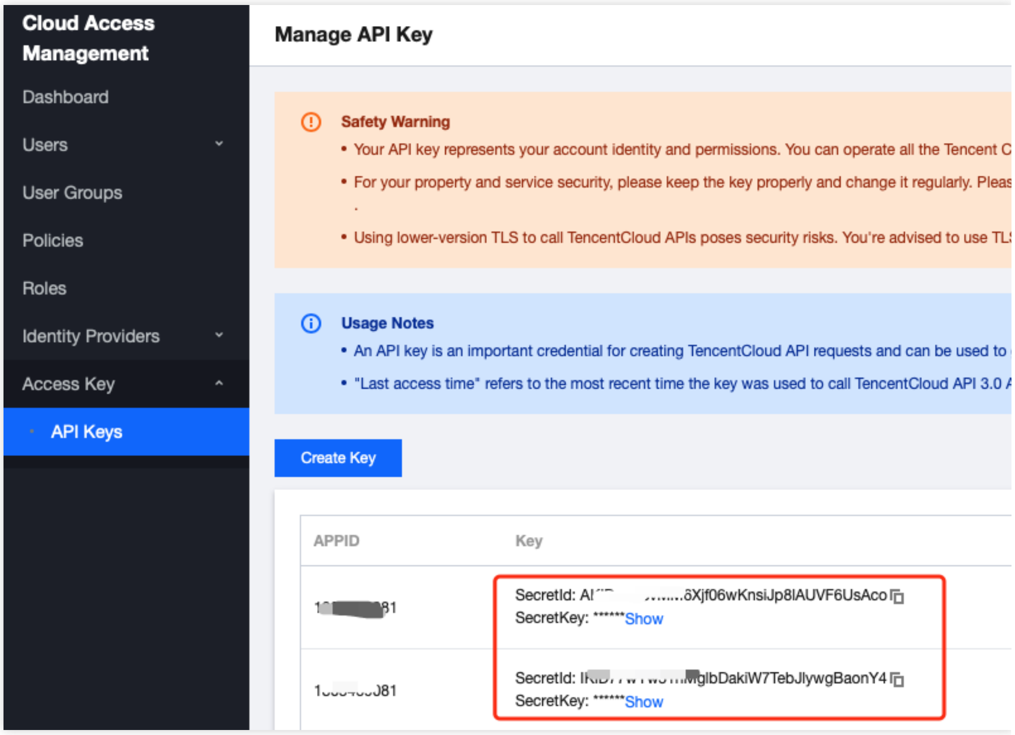
	 **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.  Cloud Access Management Dashboard Users User Groups Policies Roles Identity Providers Access Key API Keys Create Key Manage API Key Safety Warning • Your API key represents your account identity and permissions. You can operate all the Tencent C • For your property and service security, please keep the key properly and change it regularly. Pleas • Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS Usage Notes • An API key is an important credential for creating TencentCloud API requests and can be used to • "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A APPID: 1[redacted]81 — SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco; SecretKey: ******Show APPID: 1[redacted]081 — SecretId: IK[redacted]MglbDakiW7TebJlywgBaonY4; SecretKey: ******Show
queue_name	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ cc

3. Consume messages.

```
import os
import sys

sys.path.insert(0, os.path.dirname(os.path.abspath(__file__)) + "/..")

import logging
from cmq.account import Account
from cmq.cmq_exception import CMQExceptionBase
```

```
# `secretId` and `secretKey` of your Tencent Cloud account, which should be kept co
# You can get them at https://console.intl.cloud.tencent.com/cam/capi
secretId = 'AKIDSiiRtxxxx'
secretKey = 'GGzSeaM5xxxx'
# CMQ service call address
nameServerAddress = 'https://cmq-gz.public.tencenttdmq.com'

# Initialize `my_account` and `my_queue`
# Account class objects are not thread-safe. If you need to use them in multiple th
my_account = Account(nameServerAddress, secretId, secretKey, debug=True)
my_account.set_log_level(logging.DEBUG)
queue_name = sys.argv[1] if len(sys.argv) > 1 else "python_queue"
my_queue = my_account.get_queue(queue_name)

try:
     wait_seconds = 3
     # Obtain messages
     recv_msg = my_queue.receive_message(wait_seconds)
     # Specific business
     print("Receive Message Succeed! ReceiptHandle:%s MessageBody:%s MessageID:%s"
             recv_msg.receiptHandle, recv_msg.msgBody, recv_msg.msgId))
     # Successfully consumed messages are deleted
     my_queue.delete_message(recv_msg.receiptHandle)
except CMQExceptionBase as e:
     print("Receive Message Fail! Exception:%s\\n" % e)
```

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the for CMQ console.

	 **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region OK
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.  **Cloud Access Management** Dashboard Users User Groups Policies Roles Identity Providers Access Key API Keys **Manage API Key** **Safety Warning** • Your API key represents your account identity and permissions. You can operate all the Tencent C • For your property and service security, please keep the key properly and change it regularly. Pleas . • Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TL **Usage Notes** • An API key is an important credential for creating TencentCloud API requests and can be used to • "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A Create Key APPID | Key 1[redacted]81 | SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show 1[redacted]081 | SecretId: I[redacted]MglbDakiW7TebJlywgBaonY4 SecretKey: ******Show
queue	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ co

Topic Model

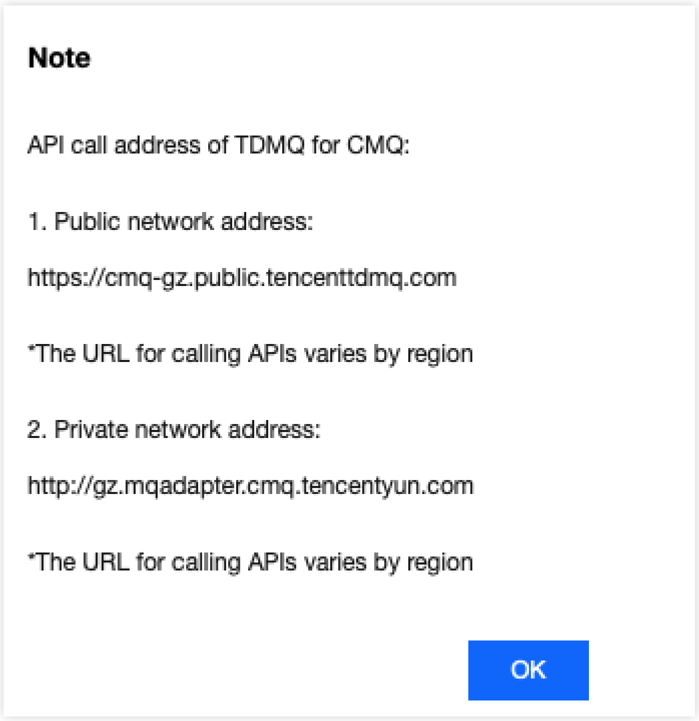
Directions

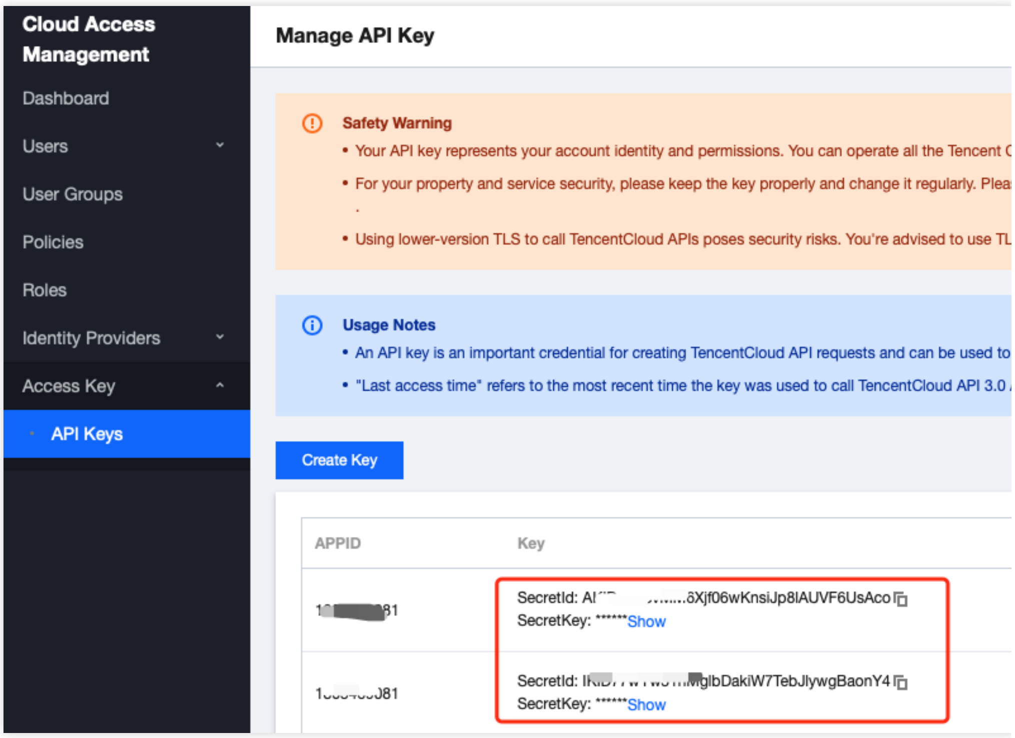
1. Prepare the required resources and create a topic subscription and a subscriber.

1.1 Create a topic subscription in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK for Python.

```
# API authentication information
cred = credential.Credential(SecretId, SecretKey)
httpProfile = HttpProfile()
```

```
httpProfile.endpoint = NameServerAddress

clientProfile = ClientProfile()
clientProfile.httpProfile = httpProfile
client = tdmq_client.TdmqClient(cred, "ap-guangzhou", clientProfile)

req = models.CreateCmqTopicRequest()
params = {
        "TopicName": "topic_api",  # Topic name, which must be unique in the same r
        "FilterType": 1,  # Used to specify the message match policy for the topic.
        "MsgRetentionSeconds": 86400  # Message retention period. Value range: 60-8
}
req.from_json_string(json.dumps(params))

# Create a topic
resp = client.CreateCmqTopic(req)
```

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region OK
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Plea
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TL

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 /

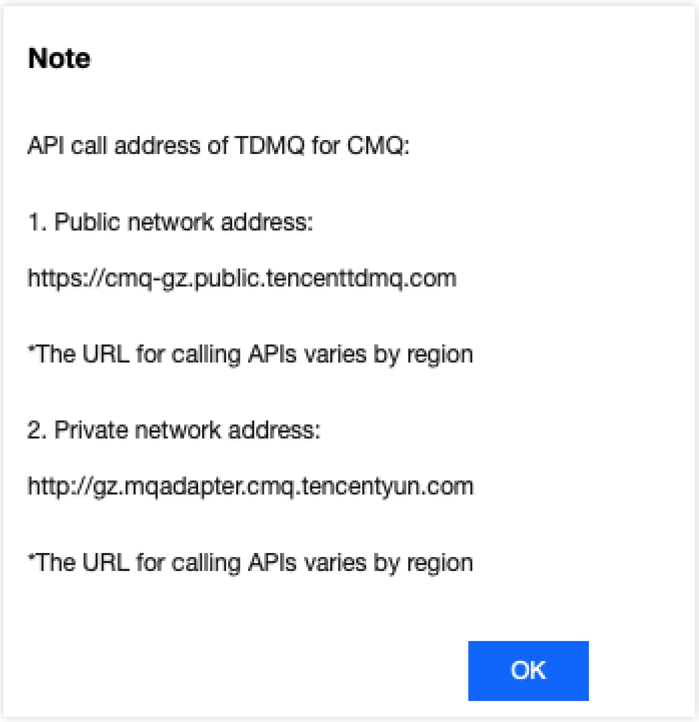
APPID	Key
1[redacted]81	SecretId: AI[redacted]8Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: IK[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

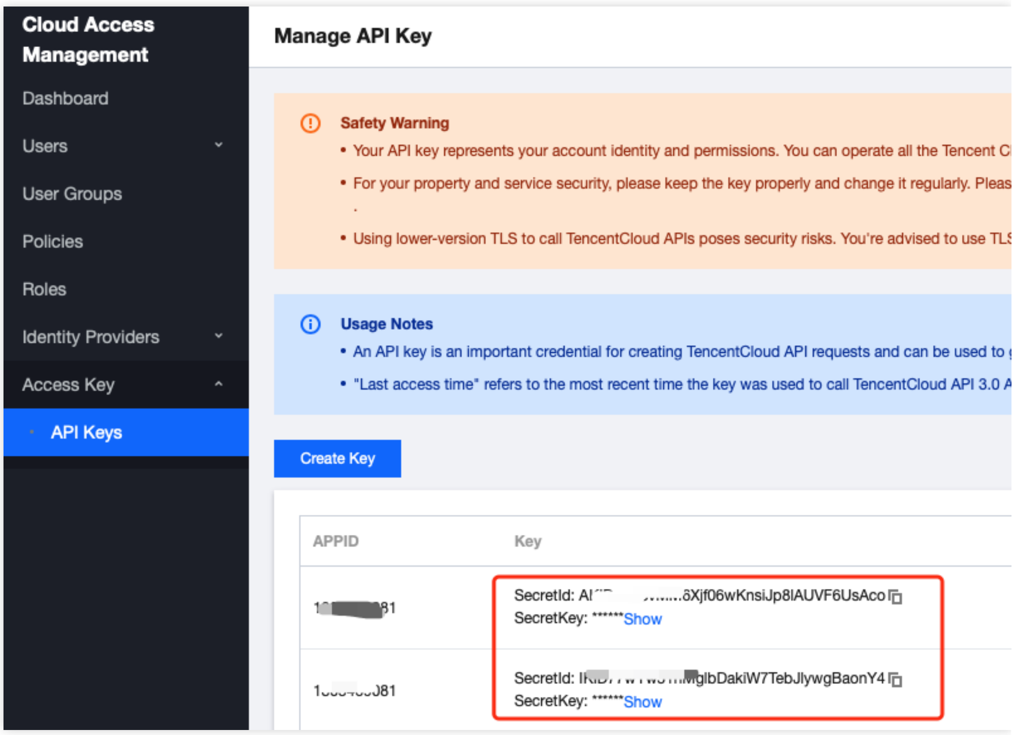
1.2 Create a subscriber in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK for Python.

```
# API authentication information
cred = credential.Credential(SecretId, SecretKey)
httpProfile = HttpProfile()
httpProfile.endpoint = NameServerAddress

clientProfile = ClientProfile()
clientProfile.httpProfile = httpProfile
client = tdmq_client.TdmqClient(cred, "ap-guangzhou", clientProfile)

req = models.CreateCmqSubscribeRequest()
params = {
        "TopicName": "topic_api",  # Name of the topic for which to create a subscr
        "SubscriptionName": "sub",  # Subscription name
        "Protocol": "queue",  # Subscription protocol. Currently, two protocols are
        "Endpoint": "topic_queue_api",   # Endpoint that receives notifications, wh
        "NotifyStrategy": "BACKOFF_RETRY",  # CMQ push server retry policy. Valid v
        "FilterTag": ["TAG"],  # Message filter tag (used for message filtering). T
        # "BindingKey": ["a.b.c"],  # The number of `BindingKey` cannot exceed 5, a
        "NotifyContentFormat": "SIMPLIFIED"  # Push content format. Valid values: 1
}
req.from_json_string(json.dumps(params))

# Create a subscription
resp = client.CreateCmqSubscribe(req)
```

Note:

`BindingKey` and `FilterTag` should be set according to the type of subscribed topic; otherwise, they will be invalid.

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

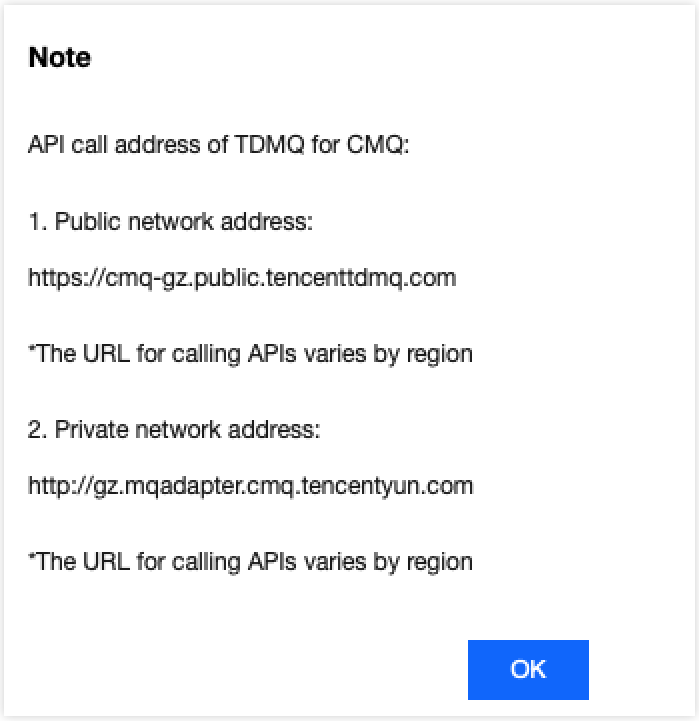
Usage Notes

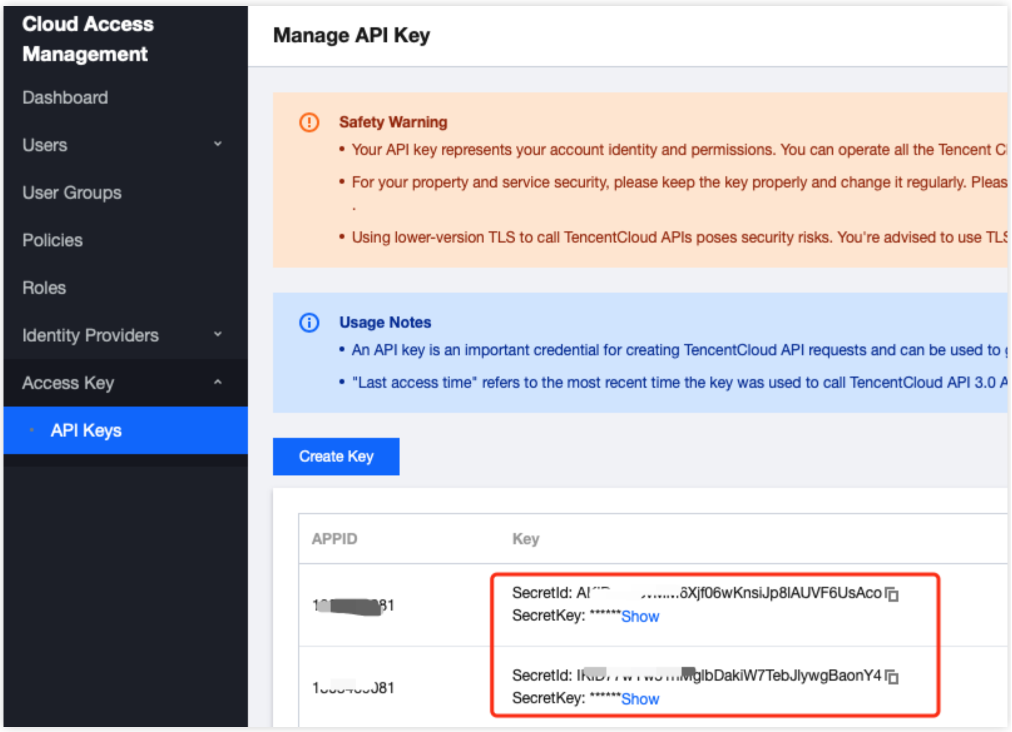
- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8IAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: IK[redacted]MglbDakiW7TebJlywgBaonY4 SecretKey: ******Show

2. Import CMQ files into the project. You need to select a branch based on the used Python version. The default SDK is for Python 2, and you can switch to the Python 3 branch to view the SDK for Python 3.
3. Create `my_topic` to publish messages.

```
import os
import sys

sys.path.insert(0, os.path.dirname(os.path.abspath(__file__)) + "/..")

import logging
from cmq.account import Account
from cmq.cmq_exception import *
from cmq.topic import *

# `secretId` and `secretKey` of your Tencent Cloud account, which should be kept co
# You can get them at https://console.intl.cloud.tencent.com/cam/capi
secretId = 'AKIDSiiRtxxxx'
secretKey = 'GGzSeaM5xxxx'
# CMQ service call address
nameServerAddress = 'https://cmq-gz.public.tencenttdmq.com'

try:
     # Initialize `my_account`
     # Account class objects are not thread-safe. If you need to use them in multip
     my_account = Account(nameServerAddress, secretId, secretKey, debug=True)
     my_account.set_log_level(logging.DEBUG)
     # Topic name
     topic_name = sys.argv[1] if len(sys.argv) > 1 else "python_topic_route"
     my_topic = my_account.get_topic(topic_name)
except CMQExceptionBase as e:
     print("Exception:%s\\n" % e)
```

Parameter	Description
NameServerAddress	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region OK
SecretId, SecretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management p the CAM console.

	
topic_name	Topic subscription name, which can be obtained on the Topic Subscription page in the TD for CMQ console.

Cloud Access Management

Dashboard

Users

User Groups

Policies

Roles

Identity Providers

Access Key

API Keys

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

Create Key

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: I[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

4. Send tag messages.

```
# Message tag
tags = ["TAG", "TAG1", "TAG2"]
for tag in tags:
        # Send tag messages
        message = Message("this is a test TAG message. TAG:" + tag, [tag])
        re_msg = my_topic.publish_message(message)
        # Sending result
        print("Send Message Succeed! MessageBody:%s MessageID:%s" % (message.msgBo
```

5. Send route messages.

```
# Message route information
routes = ["a.b.c", "a.b.x", "a.c.d", "x.y.z", "x.y.c"]
for route in routes:
    message = Message("this is a test route message. Route:" + route)
# Send route messages
re_msg = my_topic.publish_message(message, route)
# Sending result
print("Send Message Succeed! MessageBody:%s MessageID:%s" % (message.msgBody, re_ms
```

6. The consumer can consume messages in the message queue subscribed to by the subscriber.

Note:

Above is a brief introduction to message production and consumption in two models. For more information, see Demo or TDMQ for CMQ code repository.

SDK for PHP

Last updated：2024-01-03 10:20:36

Directions

This document uses the SDK for PHP as an example to describe how to connect the client to TDMQ for CMQ and send/receive messages.

Prerequisites

You have installed PHP 5.6 or later.
You have downloaded the demo.

Queue Model

Directions

1. Create a queue service in the console as instructed in Queue Management.
Note:
You can create a message queue in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK 3.0 for PHP.
2. Add dependencies.

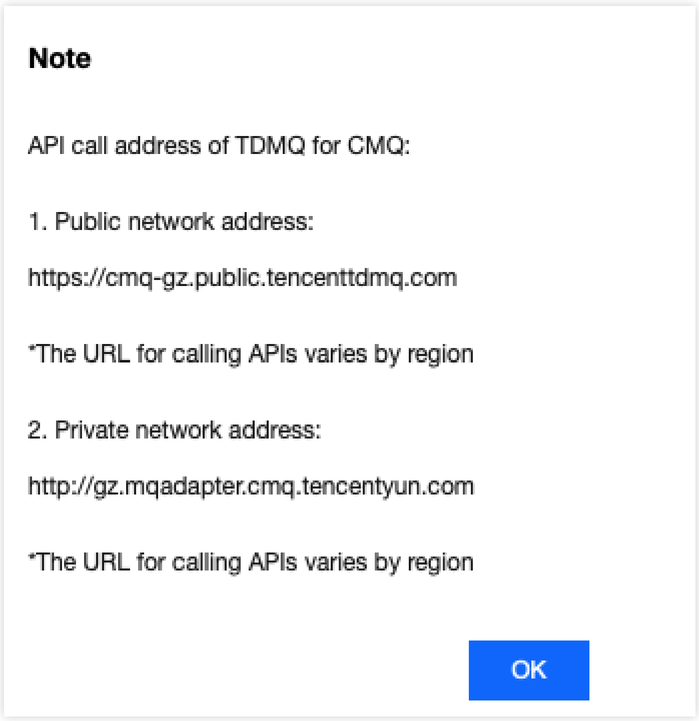
```
composer require tencentcloud/tencentcloud-sdk-php
```

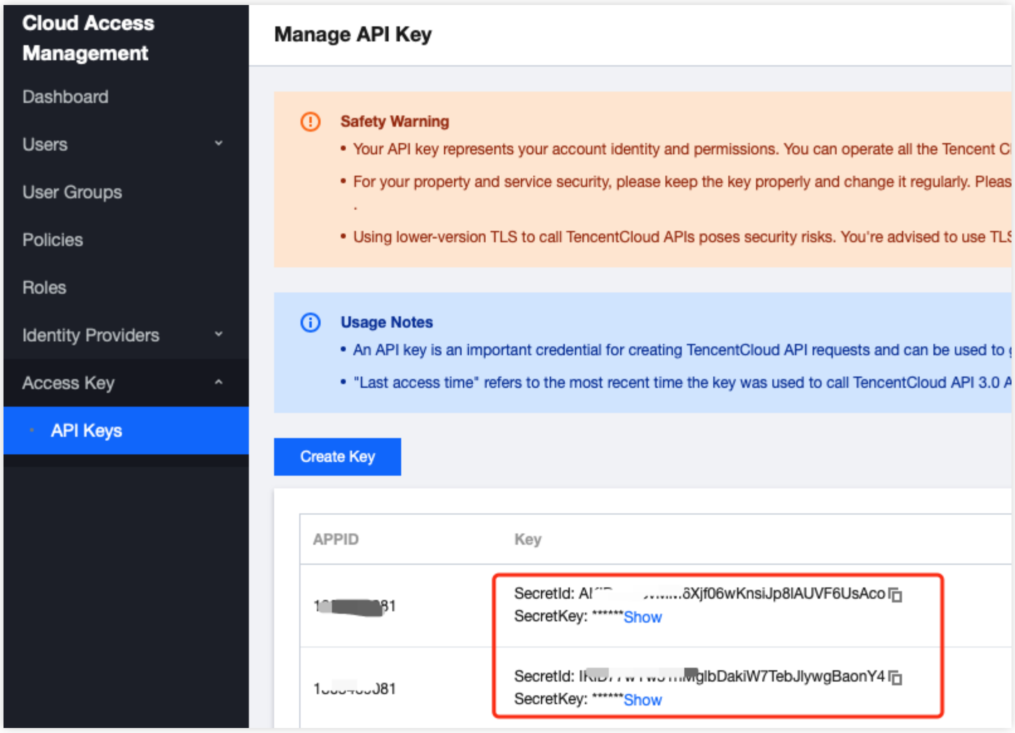
3. Add references.

```
require '/path/to/vendor/autoload.php';
```

4. Create a message queue.

```
    $cred = new Credential($secretId, $secretKey);
    $httpProfile = new HttpProfile();
    $httpProfile->setEndpoint($endPoint);

    $clientProfile = new ClientProfile();
    $clientProfile->setHttpProfile($httpProfile);
    $client = new TdmqClient($cred, "ap-guangzhou", $clientProfile);

    $req = new CreateCmqQueueRequest();

    $params = array(
        "QueueName" => "queue_api",  // Message queue name
        // Below is the dead letter queue configuration
        "DeadLetterQueueName" => "dead_queue_api", // Name of the dead letter queue
        "Policy" => 0,  // Dead letter policy. 0: Message has been consumed multiple
        "MaxReceiveCount" => 3  // Maximum receipts. Value range: 1-1000
        // MaxTimeToLive: Maximum period in seconds before an unconsumed message exp
    );
    $req->fromJsonString(json_encode($params));

    $resp = $client->CreateCmqQueue($req);
```

Parameter	Description
$endPoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
$secretId, $secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console.

Cloud Access Management

Dashboard
Users
User Groups
Policies
Roles
Identity Providers
Access Key
API Keys

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

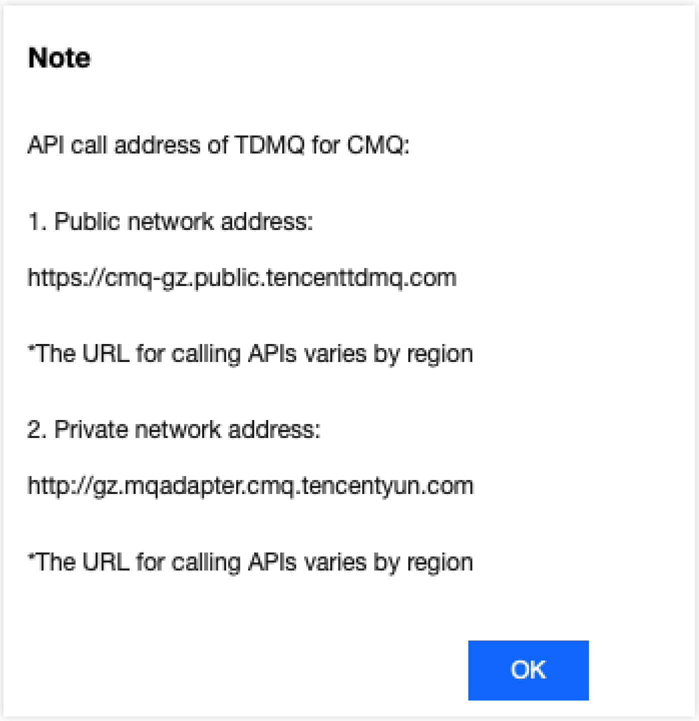
Create Key

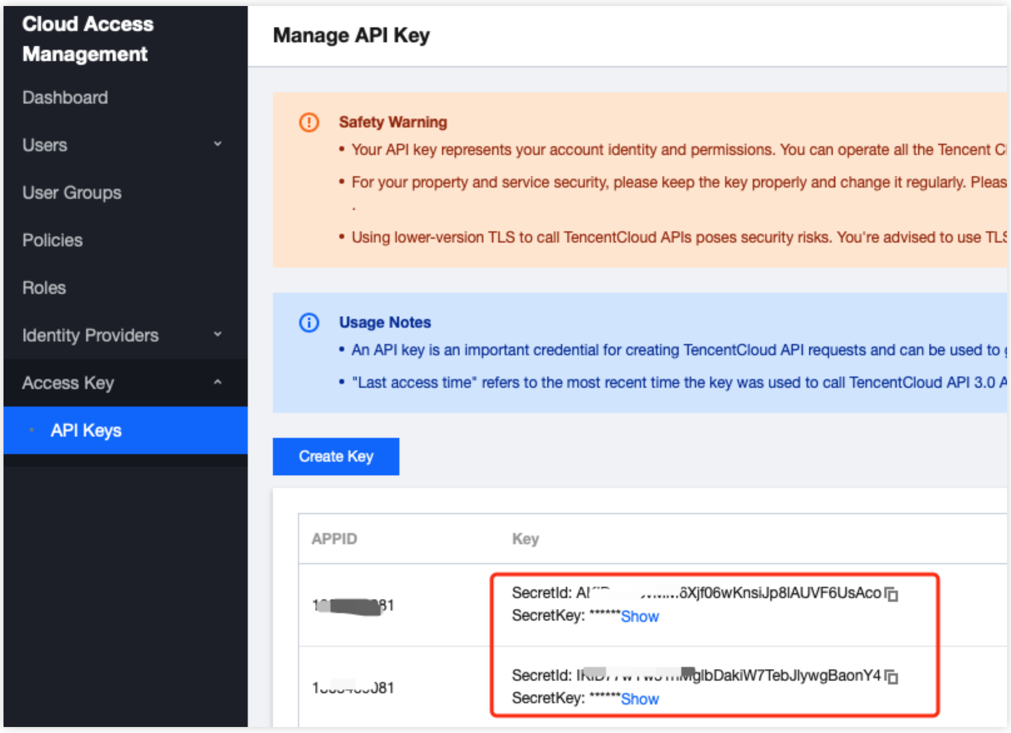
APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: I[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

5. Import CMQ files into the project.6. Send messages.

```
// Message queue name
$queue_name = "php_queue";
// Authentication information
$my_account = new Account($this->endpoint, $this->secretId, $this->secretKey);

$my_queue = $my_account->get_queue($queue_name);

$queue_meta = new QueueMeta();
$queue_meta->queueName = $queue_name;
$queue_meta->pollingWaitSeconds = 10;
$queue_meta->visibilityTimeout = 10;
```

```
$queue_meta->maxMsgSize = 1024;
$queue_meta->msgRetentionSeconds = 3600;

try {
    // Message content
    $msg_body = "I am test message.";
    $msg = new Message($msg_body);
    // Send messages
    $re_msg = $my_queue->send_message($msg);
    echo "Send Message Succeed! MessageBody:" . $msg_body . " MessageID:" . $re_msg
} catch (CMQExceptionBase $e) {
    echo "Create Queue Fail! Exception: " . $e;
    return;
}
```

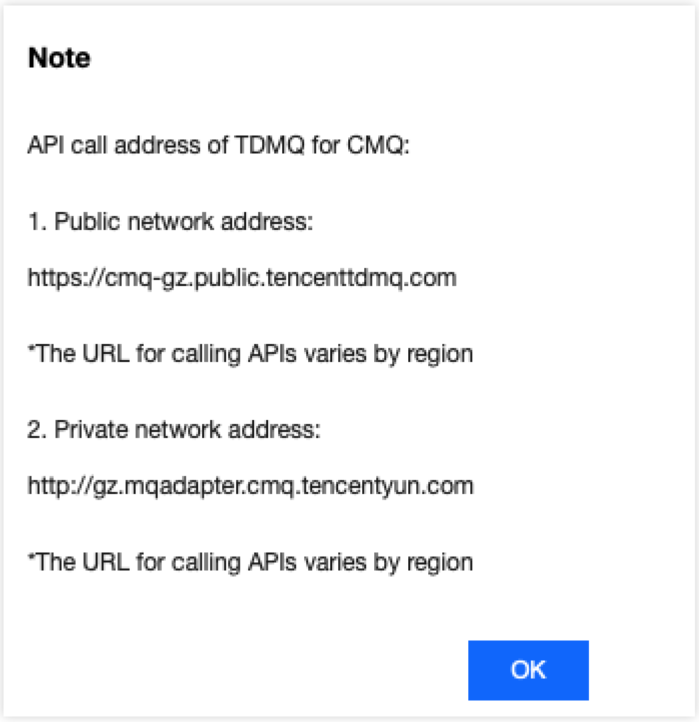
Parameter	Description
endpoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  Note API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
secretId, secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console.

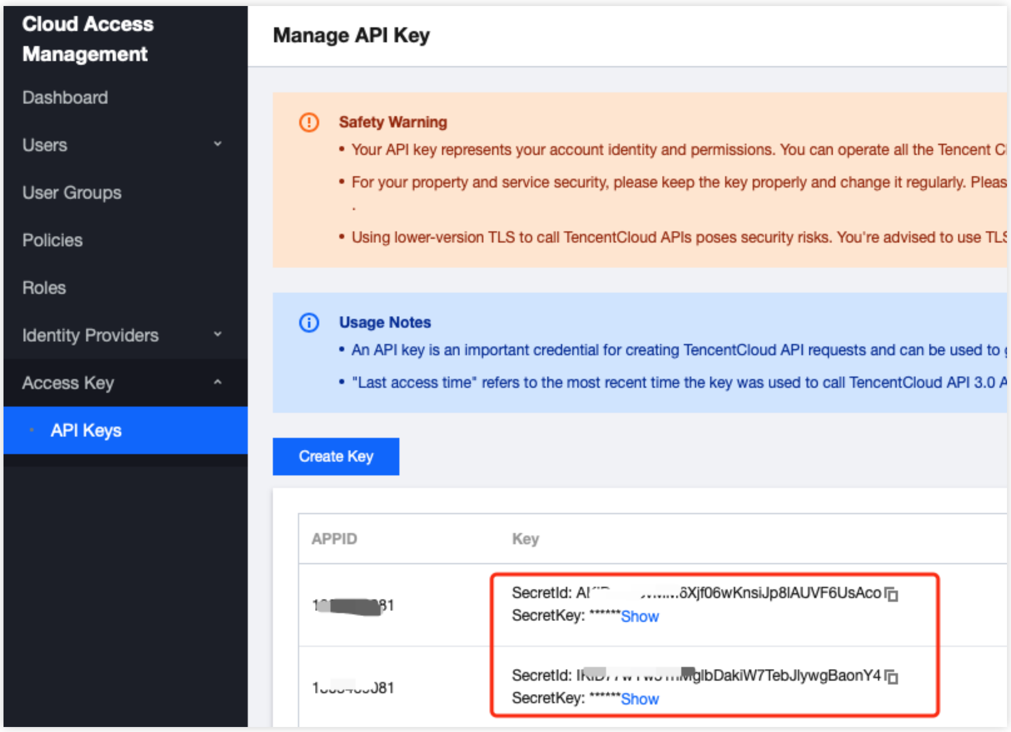
	 Cloud Access Management Dashboard Users User Groups Policies Roles Identity Providers Access Key API Keys Manage API Key Safety Warning • Your API key represents your account identity and permissions. You can operate all the Tencent C • For your property and service security, please keep the key properly and change it regularly. Pleas • Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS Usage Notes • An API key is an important credential for creating TencentCloud API requests and can be used to • "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A Create Key APPID | Key 1[redacted]81 | SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show 1[redacted]081 | SecretId: I[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show
$queue_name	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ console.

7. Consume messages.

```
// Message queue name
$queue_name = "php_queue";
// Authentication result
$my_account = new Account($this->endpoint, $this->secretId, $this->secretKey);

$my_queue = $my_account->get_queue($queue_name);

$queue_meta = new QueueMeta();
$queue_meta->queueName = $queue_name;
$queue_meta->pollingWaitSeconds = 10;
$queue_meta->visibilityTimeout = 10;
```

```
$queue_meta->maxMsgSize = 1024;
$queue_meta->msgRetentionSeconds = 3600;

try {
    // Obtain messages
    $recv_msg = $my_queue->receive_message(3);
    echo "Receive Message Succeed! " . $recv_msg . "\\n";
    // Successfully consumed messages are deleted
    $my_queue->delete_message($recv_msg->receiptHandle);
} catch (CMQExceptionBase $e) {
    echo "Create Queue Fail! Exception: " . $e;
    return;
}
```

Parameter	Description
endpoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
secretId, secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console.

	
$queue_name	Queue name, which can be obtained on the Queue Service page in the TDMQ for CMQ console.

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

Create Key

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8IAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: IK[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

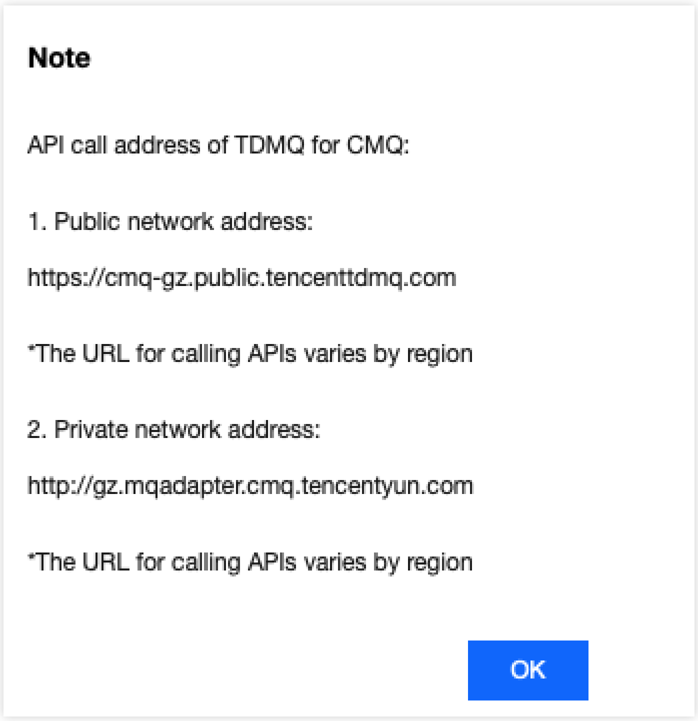
Topic Model

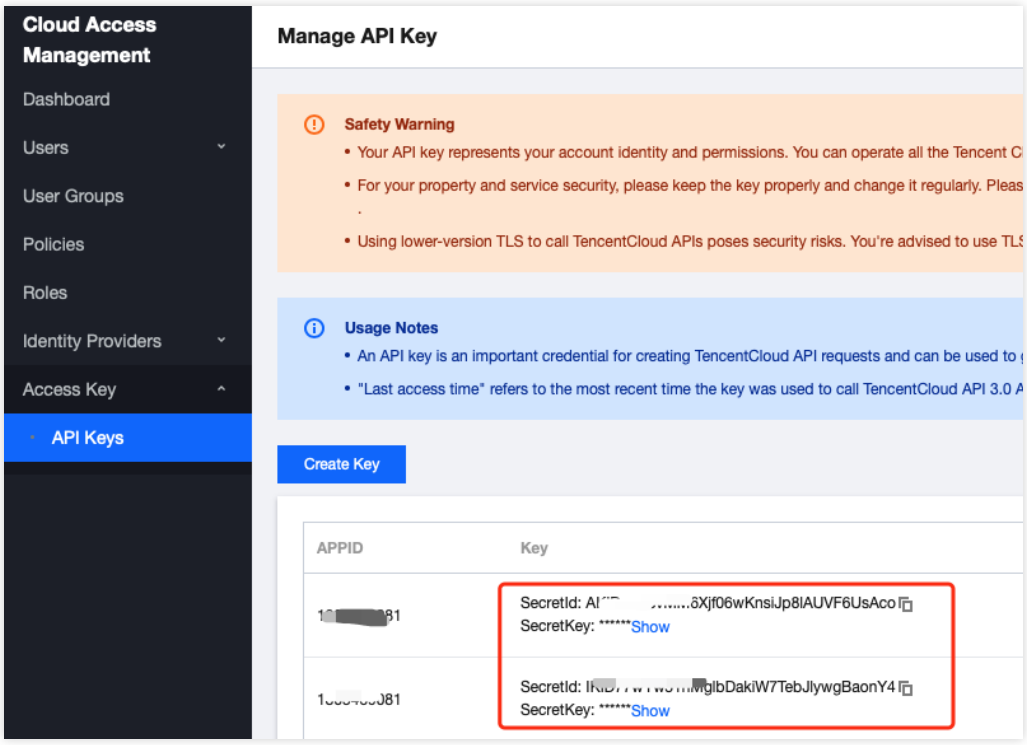
Directions

1. Prepare the required resources and create a topic subscription and a subscriber.

1.1 Create a topic subscription in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK 3.0 for PHP.

```
    $cred = new Credential($secretId, $secretKey);
    $httpProfile = new HttpProfile();
    $httpProfile->setEndpoint($endPoint);

    $clientProfile = new ClientProfile();
    $clientProfile->setHttpProfile($httpProfile);
    $client = new TdmqClient($cred, "ap-guangzhou", $clientProfile);

    $req = new CreateCmqTopicRequest();

    $params = array(
        "TopicName" => "topic_api1", // Topic name, which must be unique in the s
        "FilterType" => 2, // Used to specify the message match policy for the to
        "MsgRetentionSeconds" => 86400 // Message retention period. Value range:
    );
    $req->fromJsonString(json_encode($params));

    // Create a topic
    $resp = $client->CreateCmqTopic($req);
```

Parameter	Description
$endPoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
$secretId, $secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console.



Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

Create Key

APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: I[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

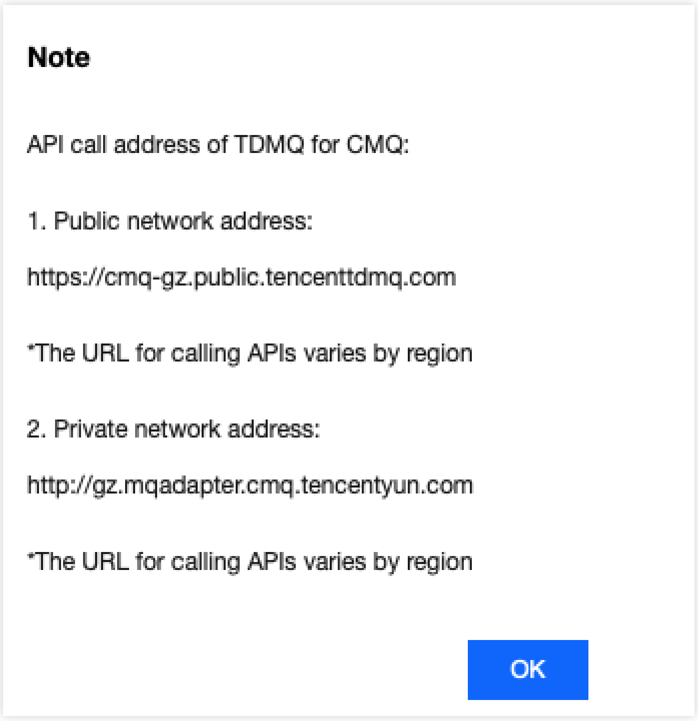
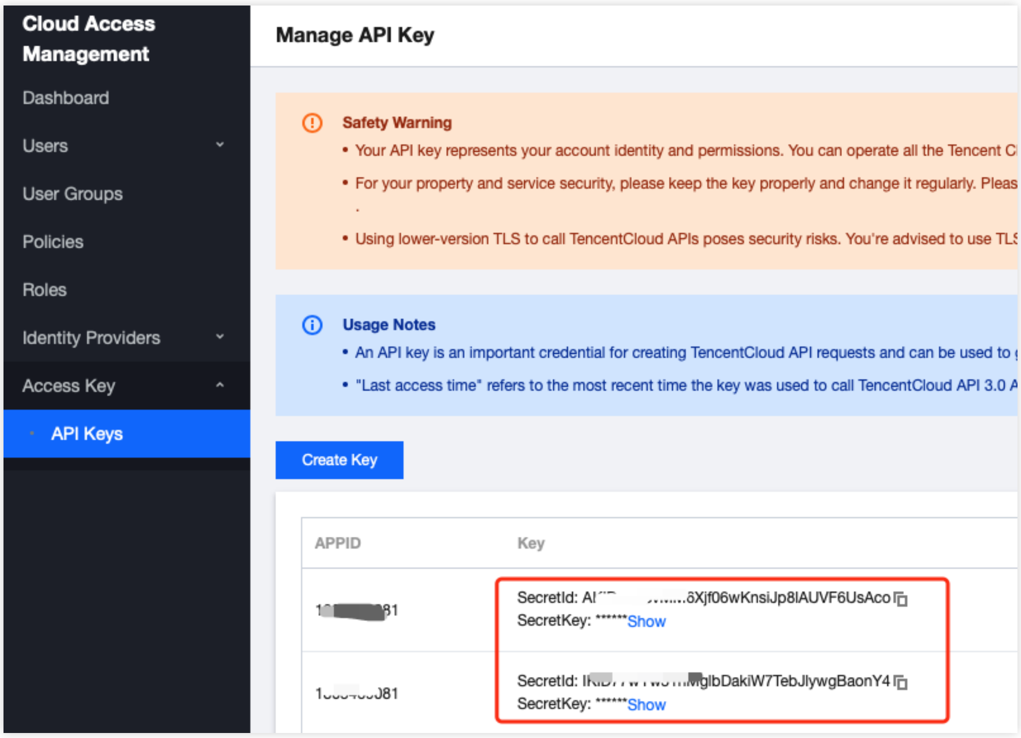
1.2 Create a subscriber in the console or via TencentCloud API. To use TencentCloud API, you need to install the SDK 3.0 for PHP.

```
        $cred = new Credential($secretId, $secretKey);
        $httpProfile = new HttpProfile();
        $httpProfile->setEndpoint($endPoint);

        $clientProfile = new ClientProfile();
        $clientProfile->setHttpProfile($httpProfile);
        $client = new TdmqClient($cred, "ap-guangzhou", $clientProfile);

        $req = new CreateCmqSubscribeRequest();

        $params = array(
            // Name of the topic for which to create a subscription
            "TopicName" => "topic_api1",
            // Subscription name
            "SubscriptionName" => "sub1",
            // Subscription protocol. Currently, two protocols are supported: HTTP an
            "Protocol" => "queue",
            // Endpoint that receives notifications, which varies by protocol: for HT
            "Endpoint" => "topic_queue_api",
            // CMQ push server retry policy. Valid values: 1) BACKOFF_RETRY; 2) EXPON
            "NotifyStrategy" => "BACKOFF_RETRY",
            // The number of `BindingKey` cannot exceed 5, and the length of each `Bi
            "BindingKey" => array("a.b"),
            // Message filter tag (used for message filtering). The number of tags ca
            // "FilterTag" => array("TAG"),
            // Push content format. Valid values: 1. JSON; 2. SIMPLIFIED, i.e., the r
            "NotifyContentFormat" => "SIMPLIFIED"
        );
        $req->fromJsonString(json_encode($params));

        // Create a subscription
        $resp = $client->CreateCmqSubscribe($req);
```

Note:

`BindingKey` and `FilterTag` should be set according to the type of subscribed topic; otherwise, they will be invalid.

Parameter	Description
$endPoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.

	
$secretId, $secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console. 

Note

API call address of TDMQ for CMQ:

1. Public network address:

https://cmq-gz.public.tencenttdmq.com

*The URL for calling APIs varies by region

2. Private network address:

http://gz.mqadapter.cmq.tencentyun.com

*The URL for calling APIs varies by region

Manage API Key

Safety Warning
- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes
- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

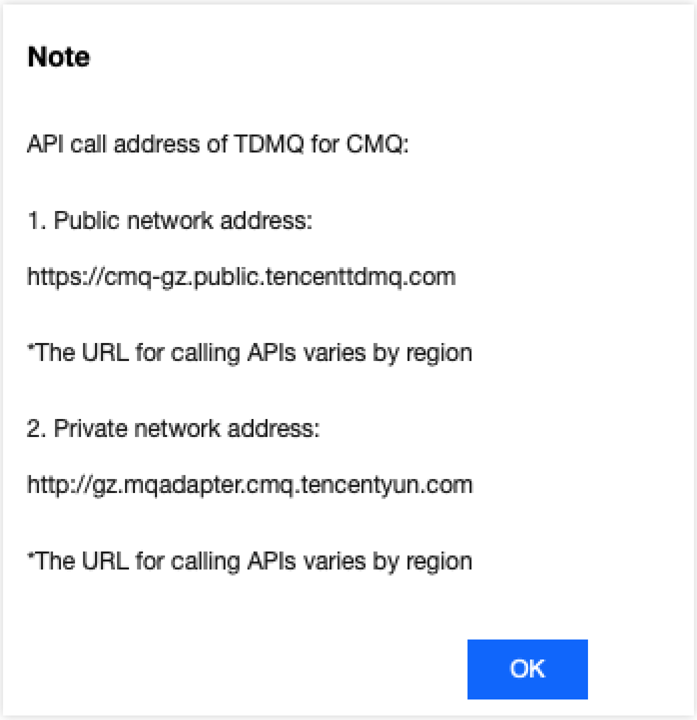
APPID	Key
1[redacted]81	SecretId: AI[redacted]6Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: IK[redacted]MglbDakiW7TebJlywgBaonY4 SecretKey: ******Show

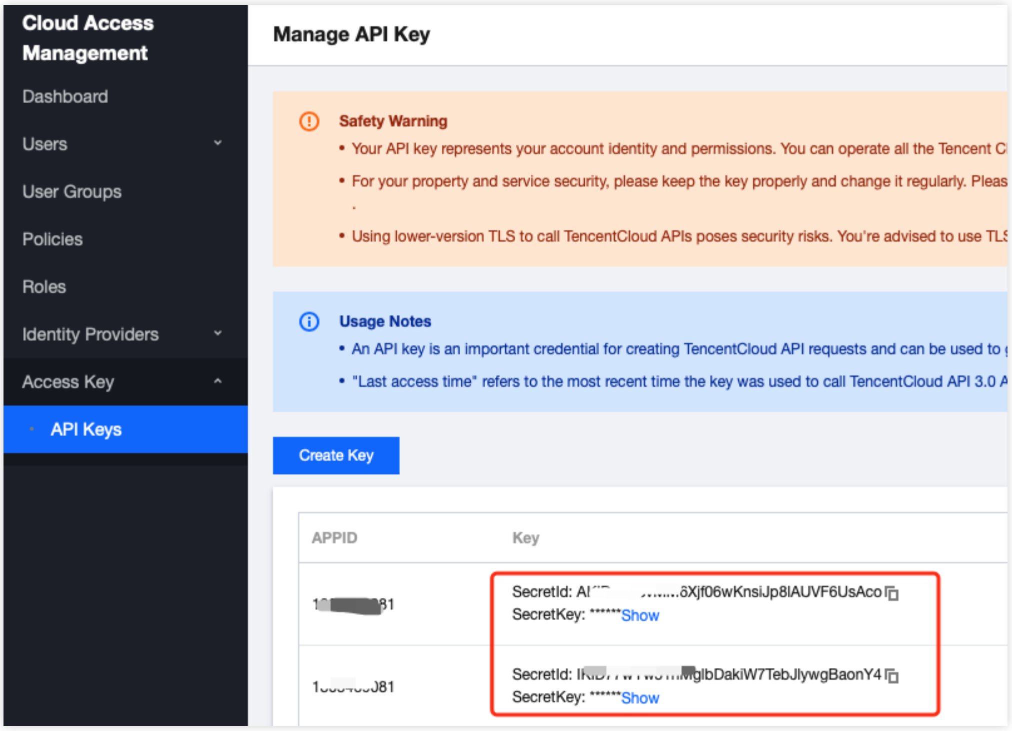
2. Import CMQ files into the project.

3. Create `my_topic` to publish messages.

```
    // Topic subscription name
    $topic_name = "php_topic_tag";
    // Authentication information
    $my_account = new Account($this->endpoint, $this->secretId, $this->secretKey);
    $my_topic = $my_account->get_topic($topic_name);
```

Parameter	Description

endpoint	API call address, which can be copied from Queue Service > API Request Address in the TDMQ for CMQ console.  **Note** API call address of TDMQ for CMQ: 1. Public network address: https://cmq-gz.public.tencenttdmq.com *The URL for calling APIs varies by region 2. Private network address: http://gz.mqadapter.cmq.tencentyun.com *The URL for calling APIs varies by region
secretId, secretKey	TencentCloud API key, which can be copied on the Access Key > API Key Management page in the CAM console.

	
$topic_name	Topic subscription name, which can be obtained on the Topic Subscription page in the TDMQ for CMQ console.

Manage API Key

Safety Warning

- Your API key represents your account identity and permissions. You can operate all the Tencent C
- For your property and service security, please keep the key properly and change it regularly. Pleas
- Using lower-version TLS to call TencentCloud APIs poses security risks. You're advised to use TLS

Usage Notes

- An API key is an important credential for creating TencentCloud API requests and can be used to
- "Last access time" refers to the most recent time the key was used to call TencentCloud API 3.0 A

APPID	Key
1[redacted]81	SecretId: AI[redacted]8Xjf06wKnsiJp8lAUVF6UsAco SecretKey: ******Show
1[redacted]081	SecretId: I[redacted]glbDakiW7TebJlywgBaonY4 SecretKey: ******Show

4. Send tag messages.

```
// Send tag messages
$msg = "this is a test message for tag.";
$msgid = $my_topic->publish_message($msg, array("TAG","TAG1"));
```

5. Send route messages.

```
// Send route messages.
$msg = "this is a test message for route.";
$msgid = $my_topic->publish_message($msg, array(), "a.b.c");
```

6. The consumer can consume messages in the message queue subscribed to by the subscriber.

Note:

Above is a brief introduction to message production and consumption in two models. For more information, see Demo or TDMQ for CMQ code repository.

HTTP Control Flow SDK

Last updated：2024-01-03 10:20:36

Overview

This document describes the control flow APIs and SDK use methods of TDMQ for CMQ. For data flow APIs (related to message sending/receiving), see HTTP Data Flow API Overview.

The control flow APIs of TDMQ for CMQ are as listed below:

API	Description
CreateCmqQueue	Creates TDMQ for CMQ queue
CreateCmqSubscribe	Creates TDMQ for CMQ subscription
CreateCmqTopic	Creates TDMQ for CMQ topic
DeleteCmqQueue	Deletes TDMQ for CMQ queue
DeleteCmqSubscribe	Deletes TDMQ for CMQ subscription
DeleteCmqTopic	Deletes TDMQ for CMQ topic
DescribeCmqDeadLetterSourceQueues	Enumerates dead letter source queues in TDMQ for CMQ
DescribeCmqQueueDetail	Queries the details of TDMQ for CMQ queue
DescribeCmqQueues	Queries all TDMQ for CMQ queues
DescribeCmqSubscriptionDetail	Queries the details of TDMQ for CMQ subscription
DescribeCmqTopicDetail	Queries the details of TDMQ for CMQ topic
DescribeCmqTopics	Enumerates all TDMQ for CMQ topics
ModifyCmqQueueAttribute	Modifies TDMQ for CMQ queue
ModifyCmqSubscriptionAttribute	Modifies TDMQ for CMQ subscription
ModifyCmqTopicAttribute	Modifies TDMQ for CMQ topic
RewindCmqQueue	Rewinds TDMQ for CMQ queue
UnbindCmqDeadLetter	Unbinds TDMQ for CMQ dead letter queue

Directions

TDMQ for CMQ's control flow SDK uses the new version of APIs. Taking **the API for creating a TDMQ for CMQ queue** as an example, the specific usage is as follows:

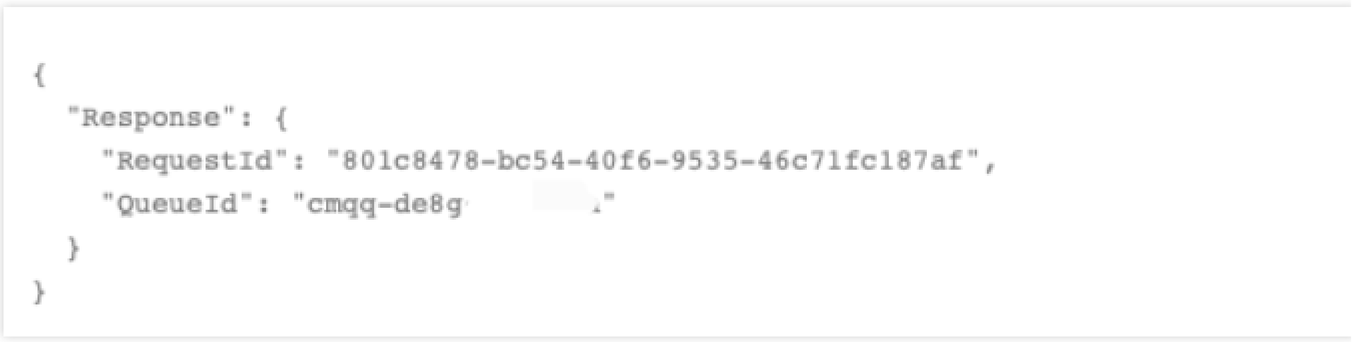
1. Log in to the TencentCloud API console.
2. Select API Explorer > Tencent Distributed Message Queue > CMQ Management APIs.
3. Select a specific API and enter the input parameters.

The input parameters are as detailed below:

Parameter	Description
QueueName	Queue name, which must be unique under the same account in the same region. It can contain up to 64 letters, digits, and hyphens and must begin with a letter.
MaxMsgHeapNum	Maximum number of heaped messages. The value range is 1,000,000–10,000,000 during the beta test and can be 1,000,000–1,000,000,000 after the product is officially released. The default value is 10,000,000 during the beta test and will be 100,000,000 after the product is officially released.
PollingWaitSeconds	Long polling wait time for message reception. Value range: 0–30 seconds. Default value: 0.
VisibilityTimeout	Message visibility timeout period. Value range: 1–43200 seconds (i.e., 12 hours). Default value: 30.
MaxMsgSize	Maximum message length. Value range: 1024–65536 bytes (i.e., 1–64 KB). Default value: 65536.
MsgRetentionSeconds	Message retention period. Value range: 60–1296000 seconds (i.e., 1 minute–15 days). Default value: 345600 (i.e., four days).
RewindSeconds	Whether to enable the message rewinding feature for a queue. Value range: 0–msgRetentionSeconds, where 0 means not to enable this feature, while `msgRetentionSeconds` indicates that the maximum rewindable period is the message retention period of the queue.
Transaction	1: transaction queue; 0: general queue.
FirstQueryInterval	First lookback interval.
MaxQueryCount	Maximum number of lookbacks.
DeadLetterQueueName	Dead letter queue name.
Policy	Dead letter policy. 0: message has been consumed multiple times but not deleted; 1: `Time-To-Live` has elapsed.

MaxReceiveCount	Maximum receipt times. Value range: 1–1000.
MaxTimeToLive	Maximum period in seconds before an unconsumed message expires, which is required if `policy` is 1. Value range: 300–43200. This value should be smaller than `msgRetentionSeconds` (maximum message retention period).
Trace	Whether to enable message trace. true: yes; false: no. If this field is not configured, the feature will not be enabled.

4. Call online.
After entering the input parameters, select the **Online Call** tab at the top of the page and click **Send Request**. The returned result is as follows:



```
{
  "Response": {
    "RequestId": "801c8478-bc54-40f6-9535-46c71fc187af",
    "QueueId": "cmqq-de8g[redacted]"
  }
}
```

5. Generate sample code.
After the verification is completed, select the target programming language in the code box on the **Code Generation** tab to generate the corresponding sample code.
Sample code for Java:

```
import com.tencentcloudapi.common.Credential;
import com.tencentcloudapi.common.profile.ClientProfile;
import com.tencentcloudapi.common.profile.HttpProfile;
import com.tencentcloudapi.common.exception.TencentCloudSDKException;
import com.tencentcloudapi.tdmq.v20200217.TdmqClient;
import com.tencentcloudapi.tdmq.v20200217.models.*;

public class CreateCmqQueue
{
    public static void main(String [] args) {
        try{
            // Instantiate an authentication object. Pass in `secretId` and `secret
            // You can get them at https://console.intl.cloud.tencent.com/cam/capi
            Credential cred = new Credential("SecretId", "SecretKey");
            // (Optional) Instantiate an HTTP option
            HttpProfile httpProfile = new HttpProfile();
            httpProfile.setEndpoint("tdmq.tencentcloudapi.com");
            // Instantiate a client option (optional; skip if no special requiremen
            ClientProfile clientProfile = new ClientProfile();
            clientProfile.setHttpProfile(httpProfile);
            // Instantiate the client object of the requested product. `clientProfi
            TdmqClient client = new TdmqClient(cred, "ap-guangzhou", clientProfile)
            // Instantiate a request object. Each API corresponds to a request obje
            CreateCmqQueueRequest req = new CreateCmqQueueRequest();
            req.setQueueName("queen");
            req.setPollingWaitSeconds(10L);
            req.setVisibilityTimeout(10L);
            req.setMaxMsgSize(1048576L);
            req.setMsgRetentionSeconds(345600L);
            // The returned `resp` is an instance of `CreateCmqQueueResponse` which
            CreateCmqQueueResponse resp = client.CreateCmqQueue(req);
            // A string response packet in JSON format is output
            System.out.println(CreateCmqQueueResponse.toJsonString(resp));
        } catch (TencentCloudSDKException e) {
            System.out.println(e.toString());
        }
    }
}
```

SDK Parameter Configuration Description

Last updated：2024-01-03 10:20:36

TDMQ for CMQ is a distributed message queue service that provides a reliable message-based async communication mechanism. It enables message sending/receiving among different applications deployed in a distributed manner (or different components of the same application) and stores the delivered messages in reliable and valid TDMQ for CMQ queues to prevent message loss. It supports multi-process simultaneous read/write, so that message sending and receiving do not interfere with each other, eliminating the need for the applications or components to keep running.
TDMQ for CMQ provides four SDKs. This document uses the SDK for Python as an example.

SDK for Python Overview

To facilitate your use, TDMQ for CMQ operations for objects such as users, queues, and topics are divided into the following classes:
Account: encapsulates account `SecretId` and `SecretKey` so that you can create, delete, and view queues, topics, and subscriptions.
queue: receives/sends messages and views/sets queue attributes.
topic: publishes messages, views/sets topic attributes, and views subscribers.
cmq_client: sets the attributes of the connection between client and server, such as setting whether log write is enabled, connection timeout period, and whether persistent connection is enabled
Note:
All classes are not thread-safe. If you need to use them in multiple threads, it is recommended that each thread instantiate its own objects.

Queue Model

A queue in TDMQ for CMQ is different from that defined in a data structure. Queues in data structures are manipulated in strict compliance with the First In, First Out (FIFO) rule, while TDMQ for CMQ distributed queues do not strictly follow FIFO (a dedicated FIFO product will be released in the future). A TDMQ for CMQ queue is considered as a container featuring high performance, capacity, and reliability, to which produced messages can be delivered, and from which messages can be fetched out for consumption. It has its own attribute settings during initialization as detailed below:

Attribute	Description
maxMsgHeapNum	Maximum number of retained messages, i.e., number of messages that can be stored

	in a queue. It represents the queue storage and retention capacity.
pollingWaitSeconds	Long-Polling waiting time for message receipt, which ranges from 0 to 30 seconds. It indicates the default time it takes to receive a message during message consumption. For example, when this attribute is set to 10, if there are no messages during message consumption, the result will be returned after 10 seconds of waiting by default; otherwise, the result will be immediately returned. You can customize the waiting time when receiving messages, which will take precedence over this attribute.
visibilityTimeout	Message visibility timeout period. After a message is obtained by a consumer, there will be an invisibility period, during which other consumers cannot get this message. This value ranges from 1 to 43,200 seconds (12 hours), and the default value is 30.
maxMsgSize	Maximum message size, which ranges from 1,024 to 1,048,576 bytes (i.e., 1–1,024 KB). The default value is 65,536.
msgRetentionSeconds	Message lifecycle, i.e., message retention time period in queue, which ranges from 60 to 1,296,000 seconds (1 minute to 15 days). The default value is 345,600 (4 days).
createTime	Queue creation time. A Unix timestamp accurate down to the second will be returned.
lastModifyTime	Time when the queue attribute is last modified. A Unix timestamp accurate down to the second will be returned.
activeMsgNum	Total number of messages in `Active` state (not being consumed) in queue, which is an approximate value.
inactiveMsgNum	Total number of messages in `Inactive` state (being consumed) in queue, which is an approximate value.
rewindSeconds	Maximum time range during which a message can be rewound in the queue, which ranges from 0 to 43,200 seconds. 0 indicates that message rewind is disabled.
rewindmsgNum	Number of retained messages which have been deleted by the `DelMsg` API but are still within their rewind time range.
minMsgTime	Minimum unconsumed time of message in seconds.
delayMsgNum	Number of delayed messages.

Getting started with the queue model >>

Topic Model

The topic model is similar to the publish/subscribe design pattern. A topic is the unit for sending messages, and subscribers under a topic are equivalent to observers. A topic will actively push published messages to subscribers.

Attribute	Description
msgCount	Number of messages currently retained in topic (number of retained messages)
maxMsgSize	Maximum message size, which ranges from 1,024 to 1,048,576 bytes (i.e., 1–1,024 KB). The default value is 65,536.
msgRetentionSeconds	Maximum lifecycle of message in topic. After the period specified by this parameter has elapsed since a message is sent to the topic, the message will be deleted no matter whether it has been successfully pushed to the user. This parameter is measured in seconds and defaulted to one day (86,400 seconds), which cannot be modified.
createTime	Topic creation time. A Unix timestamp accurate down to the second will be returned.
lastModifyTime	Time when the topic attribute is last modified. A Unix timestamp accurate down to the second will be returned.
filterType	Filtering policy selected when a subscription is created: If `filterType` is 1, `filterTag` will be used for filtering. If `filterType` is 2, `bindingKey` will be used for filtering.