Cloud Message Queue

API Document

Product Introduction

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

API Document Introduction

Last updated：2017-12-19 16:09:06

Welcome to CMQ (Cloud Message Queue).

Tencent Cloud's Cloud Message Queue (CMQ) is a distributed message queue service used for storing messages transferred between processes. It is designed to provide reliable message-based asynchronous communication service between different applications deployed in a distributed way or between different components of an application. Messages are stored in a highly reliable and available message queue, which allows multiple processes to perform read and write operations simultaneously without interfering with each other. With Tencent Cloud CMQ, messages can be transferred, without any data loss, between distributed components of applications executing different tasks. There is no need to keep every component available at all time.

The queue acts as a buffer between the data sender and the data receiver, eliminating the problems caused by the circumstances in which the data sender works faster than the data receiver or the data sender or receiver only connects to the network intermittently.

In the traditional process communication mode, the client requests a service from the server and waits for a response from the server. Such a mode has many shortcomings, for example, the likelihood of lost request in case of poor network conditions, or failed request due to timeout caused by overlong waiting of client in case of lengthy processing of server.

To deal with these problems, Tencent Cloud introduces the CMQ service for message distribution and management. With Tencent Cloud CMQ, components of an application can be separated to run independently, and message management between the components can be simplified. Any component of a distributed application can store messages in the queue, and Tencent Cloud CMQ ensures that each message is transmitted at least once and can be read and written many times. A single queue can be used by multiple distributed application components at the same time which are not required to collaborate with each other. All the components can programmatically retrieve and operate messages using CMQ APIs.

For all the supported operations, please see API Overview.

Before using these APIs, please make sure that you have a thorough understanding of CMQ Product Overview.

Note: Currently, collaborator account is not allowed to perform CMQ operations.

Queue Model

Glossary

The key terms involved in the document are as follows:

Term	Full Name	Chinese	Description
CMQ	Cloud Message Queue	Cloud Message Queue	Tencent Cloud CMQ includes queue models, topic model, and high-performance messaging services, etc.
Queue	Queue	Queue	Queue is a data model built on a "first-in-first-out" basis. Producers can concurrently add data to the tail of a queue, and consumers can concurrently pull data from the head of the queue.
Active	Active	Consumable (visible)	When being active, a message can be scrambled for by all the consumers, among whom only one consumer will succeeds in consuming the message. In this case, the message changes to a status of "inactive", and is invisible to other consumers.
Inactive	Inactive	Inconsumable (invisible)	A message with a status of "inactive" is being consumed by a consumer and other consumers cannot consume it.
(Batch)ReceiveMessage	(Batch) Receive Message	Consume Message	ReceiveMessage operation only changes the status of a message from "inactive" to "active" so that it cannot be consumed by other consumers. After being consumed, the message can only be deleted from the queue after (Batch)DeleteMessage operation is called explicitly. If the operation is not called, when the time specified by visibilityTimeout is up, the message will change back to the status of "active" and thus can be consumed by other consumers, leading to unexpected duplicate messages.

Term	Full Name	Chinese	Description
maxMsgHeapNum	Maximum Message Heap Number	Maximum number of heaped messages	In order to prevent loss of messages caused by the failure to consume the messages in time, CMA provides the message heaping function with an upper limit on the number of heaped messages. When the upper limit is reached, producers cannot add messages to the queue. After the messages are consumed and deleted, producers can add messages to the queue.
pollingWaitSeconds	Polling Wait Seconds	Long-Polling Waiting Time for Message Receipt	When a consumer wants to pull messages from a queue for consumption, there may not be data available in the queue. Perhaps the consumer does not want to go backn immediately (similar to the non-blocking mode) and wants to wait a while to see if some messages will come (similar to the blocking mode); pollingWaitSeconds is similar to the timeout in the blocking mode; once this time is up, the consumer must go back regardless of whether there is a message. If you want to consume a message in a non-blocking manner, set this value to 0.
msgRetentionSeconds	Message Retention Seconds	Message Retention Period	Messages heaped in the queue are kept within a certain time period and will be deleted from the queue if not consumed within the time limit. Once being deleted, , they cannot be consumed any longer.
receiptHandle	Receipt Handle	Message Receipt Handle	This handle is returned when a message is consumed. Only the handle which is currently consuming the message can be used to delete the message. If consumption is so lengthy that the visibilityTimeout is exceeded, and the message is consumed by another consumer, the handle through which the former consumer obtains the message will become invalid and cannot be used to delete the message.

Term	Full Name	Chinese	Description
qps throttling	QPS Throttling	qps limt	This term originally means the limit on the number of requests per second, but in practice it refers to the limit on the number of messages per second. For single-operation APIs (SendMessage, ReceiveMessage, DeleteMessage), it still refers to the number of requests per second. But for batch-opeartion APIs (BatchSendMessage, BatchReceiveMessage, BatchDeleteMessage), it is the sum of all the batch values within 1s. This limit is designed to provide users with a more consistent and fairer product. If you need to raise the limit on the number of messages, please send a request to us by submitting a ticket.

Definitions of input and response parameters

- limit and offset

> These parameters are used to control paging. For the results in a list format, if the number of entries exceeds the "limit" value, the number of returned values will be limited to the "limit" value. You can use the parameters "limit" and "offset" to control paging. "limit" indicates the maximum number of entries returned at a time, and "offset" is the offset value.
> For example, if offset = 0 & limit = 20, the 0th to the 20th entries will be returned; if offset=20&limit=20, the 20th to the 40th entries will be returned; if offset=40&limit=20, the 40th to the 60th entries will be returned, and so on.

- id.n

> Format for inputting multiple parameters at a time. Multiple parameters in such a format can be input at the same time. For example:
>
> id.0="10.12.243.21"&id.1="10.12.243.21"&id.2="10.12.243.21"&id.3="10.12.243.21"...
>
> And so on (starting with the subscript 0).

API Quick Start

You can use CMQ services through CMQ SDK(recommended; it is available in multiple languages) or by directly calling cloud APIs (this method is very inconvenient and is suitable for the users using a language other than SDK language):

1. Select a region and private/public network. The request domains for the CMQ APIs, unlike those of APIs of other cloud products, vary with the regions, and need to be selected based on the region. Each request domain name has a composition such as `cmq-queue-region.api.qcloud.com/v2/index.php` , where the "region" field needs to be replaced with the specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). If you are using a Tencent Cloud CVM, private network domain is preferred, otherwise a public network domain.

2. Create a queue. SDK (or cloud API) calls CreateQueue API.

3. Send a message. SDK (or cloud API) calls SendMessage (or BatchSendMessage) API.

4. Consume a message. SDK (or cloud API) calls ReceiveMessage (or BatchReceiveMessage) API.

5. Delete a message. SDK (or cloud API) calls DeleteMessage (or BatchDeleteMessage) API.

6. Delete a queue. SDK (or cloud API) calls DeleteQueue API.

Topic Model

Glossary

The key terms involved in the document are as follows:

Terms	English	Description
Subscriber	Subscriber	The subscriber to a service in CMQ-topic mode
Produce	Produce	The process where producer writes messages into topic
Deliver	Subscription	The process where messages under a topic is delivered to the subscriber

Terms	English	Description
Message-receiving model (PUSH)	Message-receiving model (PUSH)	TOPIC model of CMQ; PUSH mode (active pushing) is supported already
Retry strategy	NotifyStrategy	NotifyStrategy attribute of subscription, that is, the retry strategy used when an error occurs during the pushing of message to the receiver. This strategy is enabled by default. There are two options, out of which only one is allowed: a. backoff retry: retry three times, with the interval being a random value between 10s and 20s; after retry three times, the message will be discarded for the subscriber, and no retry will be made any more; b. Decay exponential retry: retry 176 times with the total retry time being 1 day and the internals being 2^0, 2^1, ..., 512, 512, ..., 512 seconds in turn. Decay exponential retry strategy is checked by default. One option must be selected from the two.
Message lifecycle	msgRetentionSeconds	The number of seconds for which the message can be retained in the TOPIC. The message will be deleted after this time has elapsed since it is sent to the queue, regardless of whether the message has been pulled or not. Default value is 86400s (one day) and cannot be modified
Maximum message size	MaxMsgSize	The maximum size (in byte) of the message body allowed to be sent to this queue. Valid value range is 1024-65536, i.e. 1K-64K
Message heap	MessageRetentionPeriod	Enabled by default. When the message of a producer has not been triggered and delivered to the subscriber or the subscriber fails to receive the message, the message will be placed into TOPIC temporarily for multiple retries. This is not configurable and the maximum heap period is 1 day
Retry verification	Status code	After messages under a TOPIC is delivered to the subscriber, an https return code of 200 indicates success

Terms	English	Description
Add subscriber tag	FilterTag	When a subscriber is added, a FilterTag can be added; after the addition of FilterTag, the subscriber can only receive messages with this filtertag. Each tag is a string with no more than 16 characters, and a maximum of 5 tags can be added for a single subscriber. As long as one of the tags can match the filter tag of topic, the subscriber can receive the message delivered under this Topic; if the message comes without any tag, the subscriber cannot receive this type of message
Add message filter tag	Messagetag	Message tag, or message type, used to differentiate message categories under Topic of a CMQ. MQ allows consumers to filer messages by tags so as to ensure that consumers will only consume the message types that are of interest to them. This feature is disabled by default. In this case, all the messages will be sent to all the subscribers. If a subscriber has set the tag, the subscriber will not receive the messages that do not match the tag. Message filter tag describes the tag for the message filtration in this subscription (only the messages with matching tags will be pushed). Each tag is a string with no more than 16 characters, and a maximum of 5 tags can be added for a single message.
Enable log trace	LoggingEnabled	Whether to enable log management function. True: enable; False: disable. When it is enabled, the original log of CMQ will be written into COS (Cloud Object Storage). And user can perform LOG aggregation query through the CMQ console to eliminate the need of building an analysis system independently

Definitions of input and response parameters

- limit and offset

> These parameters are used to control paging. For the results in a list format, if the number of entries exceeds the "limit" value, the number of returned values will be limited to the "limit" value. You can use the parameters "limit" and "offset" to control paging. "limit" indicates the maximum number of entries returned at a time, and "offset" is the offset value.
> For example, if offset = 0 & limit = 20, the 0th to the 20th entries will be returned; if

> offset=20&limit=20, the 20th to the 40th entries will be returned; if offset=40&limit=20, the 40th to the 60th entries will be returned, and so on.

- id.n

> Format for inputting multiple parameters at a time. Multiple parameters in such a format can be input at the same time. For example:
>
> id.0="10.12.243.21"&id.1="10.12.243.21"&id.2="10.12.243.21"&id.3="10.12.243.21"...
>
> And so on (starting with the subscript 0).

API Quick Start

You can use CMQ services through CMQ SDK (recommended; it is available in multiple languages) or by directly calling cloud APIs (this method is very inconvenient and is suitable for the users using a language other than SDK language):

1. Select a region and private/public network. The request domains for the CMQ APIs, unlike those of APIs of other cloud products, vary with the regions, and need to be selected based on the regions. Each request domain name has a composition such as `cmq-queue-region.api.qcloud.com/v2/index.php` , where the "region" field needs to be replaced with the specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). If you are using a Tencent Cloud CVM, private network domain is preferred, otherwise a public network domain.

2. Create a topic. SDK (or cloud API) calls CreateTopic API.

3. Modify topic attributes. SDK (or cloud API) calls SetTopicAttributes API.

4. Obtain topic list. SDK (or cloud API) calls ListTopic API.

5. Obtain topic attributes. SDK (or cloud API) calls GetTopicAttributes API.

6. Delete a topic. SDK (or cloud API) calls DeleteTopic API.

7. Publish a message. SDK (or cloud API) calls PublishMessage API.

8. Publish messages in batch. SDK (or cloud API) calls BatchPublishMessage API.

9. Deliver a Messages.

0. Create a subscription. SDK (or cloud API) calls Subscribe API.

1. Obtain subscription list. SDK (or cloud API) calls ListSubscriptionByTopic API.

2. Modify subscription attributes. SDK (or cloud API) calls SetSubscriptionAttributes API.

3. Obtain subscription attributes. SDK (or cloud API) calls GetSubscriptionAttributes API.

4. Delete a subscription. SDK (or cloud API) calls Unsubscribe API.

API Overview

Last updated：2018-09-10 11:14:21

Queue Model

Queue-related APIs

Feature	Action ID	Description
Create a queue	CreateQueue	It is used to create a new queue under the user's account.
Get queue list	ListQueue	It is used to display the queue list under the account, and the list can be obtained by page.
Get queue attributes	GetQueueAttributes	It is used to get attributes of a created queue.
Modify queue attributes	SetQueueAttributes	It is used to modify attributes of a message queue.
Delete a queue	DeleteQueue	It is used to delete a created queue.

Message-related APIs

Feature	Action ID	Description
Send a message	SendMessage	It is used to send a message to a specified queue.
Send a batch of messages	BatchSendMessage	It is used to send a batch of messages to a specified queue.
Consume a message	ReceiveMessage	It is used to consume a message in the queue.
Consume a batch of messages	BatchReceiveMessage	It is used to consume multiple messages in the queue.
Delete a message	DeleteMessage	It is used to delete a message that has been consumed.
Delete a batch of messages	BatchDeleteMessage	It is used to delete a batch of messages that have been consumed.

Topic Model

Topic-related APIs

Feature	Action ID	Description
Create a topic	CreateTopic	It is used to create a new topic under the user's account.
Modify topic attributes	SetTopicAttributes	It is used to modify attributes of a created topic.
Get topic list	ListTopic	It is used to display the topic list under the account, and the list can be obtained by page.
Get topic attributes	GetTopicAttributes	It is used to get attributes of a created topic.
Delete a topic	DeleteTopic	It is used to delete a created topic.

Message-related APIs

Feature	Action ID	Description
Publish a message	PublishMessage	It is used to publish a message on a specified topic.
Publish a batch of messages	BatchPublishMessage	It is used to publish a batch of messages on a specified topic.
Deliver messages		

Subscription-related APIs

Feature	Action ID	Description
Create a subscription	Subscribe	It is used to create a new subscription under the user's account.
Get subscription list	ListSubscriptionByTopic	It is used to display the subscription list under the topic, and the list can be obtained by page.
Modify subscription attributes	SetSubscriptionAttributes	It is used to set attributes of a created subscription.

Feature	Action ID	Description
Get subscription attributes	GetSubscriptionAttributes	It is used to get attributes of a created subscription.
Delete a subscription	Unsubscribe	It is used to delete a created subscription.

Update History

Last updated：2017-12-19 16:10:33

Time	Updates
Aug 8, 2016	Create a queue model document.
Dec 12, 2016	Create a topic model document.

Cases

Last updated：2018-05-15 11:03:37

In order to help you get started with Cloud Message Queue (CMQ) API quickly, we provide an example on how to use it. This article provides the guide on how to use the API to create a queue, send messages, consume messages, delete messages, and delete queues.

> Note: The following parameters are examples, and are for reference only. The actual value is subject to the return value of the system. For the sake of simplicity, we take single operations as the examples of message operations, but batch API (such as Send Messages in Batch, Delete Messages in Batch) is also supported.

Domain for public network API request: cmq-queue-region.api.qcloud.com

Domain for private network API request: cmq-queue-region.api.tencentyun.com

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.

Queue Model

Take private network as an example (for public network, replace "tencentyun" in the domain with "qcloud"). The composition of request domain is `cmq-queue-region.api.tencentyun.com/v2/index.php` , where the region field should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The region value in the Common Parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.

For more information about the generation rules of parameter Signature, refer to Signature Method.

Create a Queue

Before creating a queue, please refer to the instructions of Create Queue API, and adjust the attribute values of the queue according to your business needs.
If we create a queue in Guangzhou and the expected business messages are faster than production, the required request parameters are shown in the table below:

Parameter Name	Description	Example Value
queueName	The queue name must be unique under the same account in the same region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.	test-queue-1
pollingWaitSeconds	Waiting time for messages to be received when using long-polling. The value ranges from 0 to 30 seconds. Default is 0.	30

By combining common request parameters and API request parameters, you can get the final request as follows:

```
http://cmq-queue-gz.api.tencentyun.com/v2/index.php?
Action=CreateQueue
&Region=gz
&Timestamp=1465750149
&Nonce=46364
&SecretId=AKIDxxxxugEY
&Signature=5umi9gUWpTTyk18V2g/Yi56hqls=
&queueName=test-queue-1
&pollingWaitSeconds=30
```

The returned result of the above request is as follows:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"queueId":"queue-ajksdfasdowe"
}
```

We successfully created a queue. Users can manage the queue, send and consume messages, and perform other operations.

Send a Message

Before sending a message, please refer to the instructions of Send Message API.

If we send a message to the `test-queue-1` queue created in the example above, the required request parameters are shown in the table below:

Parameter Name	Description	Example Value

Parameter Name	Description	Example Value
queueName	The queue name must be unique under the same account in the same region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.	test-queue-1
msgBody	Message text. The size is at least 1 Byte, and the maximum length is limited by the set maximum length attribute of the queue message.	This'is test message (After encoded with URL, it should be This'is+test+message)

The request is as follows:

```
http://cmq-queue-gz.api.tencentyun.com/v2/index.php?
Action=SendMessage
&Region=gz
&Timestamp=1465750149
&Nonce=46365
&SecretId=AKIDxxxxugEY
&Signature=5umi9gUWagTTyk18V2g/Yi56hqls=
&queueName=test-queue-1
&msgBody=This'is+test+message
```

The returned result of the above request is as follows:

```
{
"code" : 0,
"message" : "",
"requestId":"145346456555",
"msgId":"123345346"
}
```

Consume a Message

Before consuming a message, please refer to the instructions of Consume Message API. You can choose the value of pollingWaitSeconds according to your business features.
If we consume a message from the `test-queue-1` queue created in the example above, the required request parameters are shown in the table below:

Parameter Name	Description	Example Value

Parameter Name	Description	Example Value
queueName	The queue name must be unique under the same account in the same region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.	test-queue-1
pollingWaitSeconds	Waiting time for the request when using long-polling. The value ranges from 0 to 30 seconds. Default is 0.	10

The request is as follows:

```
http://cmq-queue-gz.api.tencentyun.com/v2/index.php?
Action=ReceiveMessage
&Region=gz
&Timestamp=1465750150
&Nonce=46368
&SecretId=AKIDxxxxugEY
&Signature=5umi9gUaagTTyk18V2g/Yi56hqls=
&queueName=test-queue-1
&pollingWaitSeconds=10
```

The returned result of the above request is as follows:

```
{
"code" : 0,
"message" : "",
"requestId":"145346635355",
"msgBody":"This is test message",
"msgId":"123345346",
"receiptHandle": "283748239349283",
"enqueueTime": 1462351990,
"firstDequeueTime": 1462352990,
"nextVisibleTime": 1462352999,
"dequeueCount": 1
}
```

Delete a Message

In general, the message that has been consumed once should be deleted, unless your business has the demand for repeated consumption. Before deleting a message, please refer to the instructions of Delete Message API.

If we consume a message from the queue `test-queue-1` and delete it after consuming, please be sure to delete it before the time of nextVisibleTime. Otherwise, the receiptHandle will become invalid, and the deletion operation will fail. The required request parameters are shown in the table below:

Parameter Name	Description	Example Value
queueName	The queue name must be unique under the same account in the same region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.	test-queue-1
receiptHandle	The unique message handle returns after each consumption, which is used to delete messages. Only the message handle generated when the message was consumed last time can be used to delete this message.	"283748239349283" (receiptHandle in the example above)

The request is as follows:

```
http://cmq-queue-gz.api.tencentyun.com/v2/index.php?
Action=DeleteMessage
&Region=gz
&Timestamp=1465750151
&Nonce=46369
&SecretId=AKIDxxxxugEY
&Signature=5umi9gUaagTasdfk18V2g/Yi56hqls=
&queueName=test-queue-1
&receiptHandle=283748239349283
```

The returned result of the above request is as follows:

```
{
"code" : 0,
"message" : "",
"requestId":"14534454555"
}
```

Delete a Queue

Before deleting a queue, please refer to the instructions of Delete Queue API. When the queue is no longer used, it needs to be deleted. The required request parameters are shown in the table below:

Parameter Name	Description	Value
queueName	The queue name must be unique under the same account in the same region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.	test-queue-1

The request is as follows:

```
http://cmq-queue-gz.api.tencentyun.com/v2/index.php?
Action=DeleteQueue
&Region=gz
&Timestamp=1465750152
&Nonce=46370
&SecretId=AKIDxxxxugEY
&Signature=5umi9gUaagTasasdl18V2g/Yi56hqls=
&queueName=test-queue-1
```

The returned result of the above request is as follows:

```
{
"code" : 0,
"message" : "",
"requestId":"14534454555"
}
```

Calling Methods
Request Structure
Public Request Parameters

Last updated：2018-05-15 11:04:15

The calls to Tencent Cloud APIs are achieved by sending requests to the server IP addresses of these APIs and adding relevant request parameters in the requests as described in API descriptions. A request for calling Tencent Cloud API is made up of the following elements:

1. Service Address

The service connection address of Tencent Cloud APIs depends on the modules. For more information, please see descriptions of APIs.

2. Communication Protocol

Most of Tencent Cloud APIs make communication over HTTPS to provide high-security channels.

3. Request Method

Tencent Cloud APIs support both POST and GET requests.
**Note:

1. The two methods cannot be used at the same time. If GET method is used, parameters are obtained from the Querystring. If POST method is used, parameters are obtained from the Request Body, and the parameters in the Querystring will be ignored. The rules for parameter formats are the same for both methods. Generally, POST method is used. Because GET may encounter unexpected problems if the parameter strings are too long.
2. Regardless of which request method is used by the user, all parameters need to be encoded.**

4. Request Parameters

Two types of parameters are needed for each request for Tencent Cloud APIs - common request parameters and API request parameters. Common request parameters are the parameters common to all APIs (For more information, refer to Common Request Parameters section), while API request parameters are parameters specific to each API (For more information, refer to "Request Parameters" description of each API.)

5. Character Encoding

All requests for Tencent Cloud APIs and returned results use UTF-8 character set for encoding.

Common Request Parameters

Last updated：2018-08-14 18:10:50

Common request parameters are the parameters common to all APIs, and will not be discussed in the document for each API unless it is necessary to do so. **They are required in each request for the request to be initiated normally**. The first letter of each common request parameter is uppercase so that the parameter can be differentiated from API request parameters.

Common request parameters are listed as follows:

Name	Type	Description	Required
Action	String	The name of the API for the desired operation. For example, if you want to call the Query List of Instances API, the Action parameter is DescribeInstances.	Yes
Region	String	Region parameter, used to identify the region to which the instance you want to operate belongs. The parameter values for regions are as follows: Beijing: bj, Guangzhou: gz, Shanghai: sh, Hong Kong: hk, North America: ca. **Note: Normally this parameter is required. Otherwise it will be mentioned in the corresponding API.**	No
Timestamp	UInt	The current UNIX timestamp that records the time at which the API request was initiated.	Yes
Nonce	UInt	A random positive integer that is used in conjunction with Timestamp to prevent replay attacks.	Yes
SecretId	String	The SecretId applied for from Cloud API Key, used for identification. A SecretId corresponds to a unique SecretKey, and the SecretKey is used to generate the request Signature. For more information, refer to Signature Method.	Yes
Signature	String	Request signature, used to verify the legitimacy of the request and automatically generated by system according to input parameters. For more information, refer to Signature Method.	Yes

If a user wants to query the CVM instance list for Guangzhou, the request link may be as follows:

```
https://cvm.api.qcloud.com/v2/index.php?
Action=DescribeInstances
&SecretId=xxxxxxx
```

```
&Region=gz
&Timestamp=1465055529
&Nonce=59485
&Signature=mysignature
&<API Request Parameter>
```

A complete request needs two types of request parameters: common request parameters and API request parameters. Only six common request parameters are listed above. For information on API request parameters, refer to API Request Parameters section.

API Request Parameters

Last updated：2018-05-15 11:05:17

API request parameters are specific to each API. This means that different APIs support different API request parameters. The first letter of each API request parameter is lowercase so that the parameter can be differentiated from common request parameters.
Take the API Query List of Instances(DescribeInstances) as an example. It supports the following API request parameters:

Parameter Name	Required	Type	Description
instanceIds.n	No	String	Array of IDs of CVM instances to be queried. The array subscript starts with 0. You can use instanceId and unInstanceId. The unified resource ID unInstanceId is recommended.
lanIps.n	No	String	Array of private IPs of CVMs to be queried.
searchWord	No	String	CVM alias set by the user.
offset	No	Int	Offset; default is 0.
limit	No	Int	The maximum number of servers allowed to be queried at a time. Default is 20, and the maximum is 100.
status	No	Int	Status of CVM to be queried.
simplify	No	Int	Obtain non-real time data if simplify=1.
zoneId	No	Int	ID of availability zone. If it is left empty, the CVM instances of availability zones will be queried. To specify availability zones, you can call API Query Availability Zones(DescribeAvailabilityZones) to query availability zones.

The description of each field is as follows:

Parameter Name	The name of request parameter supported by the API, which the user can use as an API request parameter when using this API. Note: When a parameter name ends with ".n", it means the parameter is an array, and you need to input the array parameters in sequence when using it. For example, in the API "Query List of Instances" (DescribeInstances), if you input the parameter instanceIds.0=ins-0hm4gvho&instanceIds.1=ins-0hm4gvho, only CVM instances with IDs of ins-0hm4gvho and ins-0hm4gvho will be queried.

Required	Indicate whether this parameter is mandatory. "Yes" means the parameter is mandatory for the API, while "No" means the parameter is not mandatory. In the API "Query List of Instances" (DescribeInstances), all the API request parameters are not required, so the API call can be achieved simply by using common request parameters.
Type	The data type of the API parameter.
Description	A brief description of the API request parameter.

If a user wants to query the scaling group list, the request link may be as follows:

```
https://cvm.api.qcloud.com/v2/index.php?
&<Common request parameters>
&instanceIds.0=ins-0hm4gvho
&instanceIds.1=ins-8oby8q00
&offset=0
&limit=20
&status=2
&zoneId=100003
```

A complete request needs two types of request parameters: common request parameters and API request parameters. Only API request parameters are listed here. For information on common request parameters, refer to Common Request Parameters section.

Final Request Mode

Last updated：2018-05-15 11:05:50

The final request URL is made up of the following elements:
1) Request domain: The request domain for Query List of Instances(DescribeInstances) is cvm.api.qcloud.com. The actual request domain varies depending on the module to which the API belongs. For more information, refer to descriptions of APIs.
2) Request path: The request path of Cloud API is always /v2/index.php.
3) Final request parameter string: By combining common request parameters and API request parameters, you can get the final form of request URL as follows:

```
https:// + request domain + request path + ? +final request parameter string
```

The final request URL is as follows. The first six parameters are common request parameters, and the last six ones are API request parameters.

```
https://cvm.api.qcloud.com/v2/index.php?
Action=DescribeInstances
&SecretId=xxxxxxx
&Region=gz
&Timestamp=1465055529
&Nonce=59485
&Signature=mysignature
&instanceIds.0=ins-0hm4gvho
&instanceIds.1=ins-8oby8q00
&offset=0
&limit=20
&status=2
&zoneId=100003
```

Request Result

Correct Result

Last updated：2018-05-15 11:06:12

If the API call succeeds, the error code in the returned result will be 0, the error message field will be empty, and the returned data result will be displayed.

Example:

```
{
"code": 0,
"message": "",
<Returned result>
}
```

Wrong Result

Last updated：2018-06-29 18:15:47

If the API call fails, the error code in the returned result is not 0, the message field displays the detailed error information. Users can query detailed error information from the Error Codes page based on code and message.

Example of returned error:

```
{
"code": 5100,
"message": "(100004) incorrect projectId",
}
```

Error Codes

Last updated：2018-05-15 11:07:01

Error Codes

Message field indicates a module-related error.
Example:
"message": "(100004) projectId is incorrect"
It consists of two parts - the string within () indicates the module error code, and the string following () is the error description.
Different modules may produce different errors. The user can identify the cause of error based on error description. The following table lists a number of common error codes. If you do not find the error code you want, please refer to the specific API description in API document.

Common Error Code

Error Code	Module Error Code	Error Message	Description
4000	10,000	invalid request parameters	The request parameters are invalid. This error is usually caused by the fact that the parameters are not set according to the API descriptions of CMQ Cloud API.
4000	10010	lacked of required parameters	Lacked of required parameters. Users should refer to the API descriptions of CMQ Cloud API to find which required parameters are missing.
4100	10030	authentication failed	Authentication failed. Possible reasons are as follows 1) The secret key that generated the signature algorithm is incorrect; 2) The signature algorithm is incorrect (users try to implement the signature algorithm by themselves instead of using the official sdk).
4300	10040	charge overdue	Charge overdue.
6000	10050	server internal error	Internal server error. Generally, there will be a requestId returned to the user. Users can submit a ticket according to requestId to the technicians to locate the problem.

4100	10080	secret id status error	Secret id status error. The secret id may be disabled.
4000	10110	request parameters error	Request parameters error. This error is usually caused by the fact that the parameters are not set according to the API descriptions of CMQ Cloud API.
4420	10250	qps throttling	The qps reaches the maximum value, so it is restricted. In order to ensure the stability of background and call latency, the current maximum number of messages per second is 5,000. For customers with special requirements, customers can can submit a ticket for a larger one.
4100	10270	secret id is not existed	The secret id does not exist.
4000	10280	action is not existed	The parameter Action does not exist.
4000	10310	error: parameter key format error"	The parameter format is incorrect.
4000	10320	no such parameter	There is no such parameter.
4000	10330	parameter is NOT a repeatable parameter	The parameter is not a repeatable parameter. For example, in the batch APIs, BatchSendMessage and msgBody are repeatable parameters. In BatchDeleteMessage, receiptHandle is a repeatable parameter. If a parameter other than the repeatable parameters specified by each API is passed in the form of repeatable parameters, it will be treated as an error.
4000	10350	parameter value or length is out of range	Parameter value or length is out of range. For integers, this is usually a value error. For String parameters, it is a length error.
4000	10360	parameter error type	The parameter type is incorrect. For example, fill the integer as a string.
4000	10370	parameter batch size is more than 16	The batch value of batch API is more than 16.
4000	10380	parameter is	The subscripts of repeatable parameters are not consequent.

		not consequent	
4000	10390	lacked of required parameter	Lacked of required parameter.
4000	10400	cannot find parameter in uri	Cannot find parameter in uri.
4000	10410	unexpected http method, only GET or POST is supported	Unexpected http method. Only GET or POST is supported. Due to the length restriction of GET, it is recommended to use POST.
4000	10420	cannot parse	Cannot parse http message.
4000	10430	action name is not existed	The API name specified by Action does not exist.
4000	10440	account illegal, it may be an assistant account	Account illegal. It may be an assistant account. Currently, CMQ does not support the operations of assistant account.
4000	10450	secret id doesn't begin with AKID account	The secret id doesn't begin with AKID account.
4480	10460	exceed interface frequency limit, please slow down	To protect the back-end system, the call frequency limit of control APIs (such as CreateQueue, DeleteQueue, etc.) is much lower than QPS. Therefore, when you see this error, please reduce the call frequency of control APIs.

Queue Common Error Codes

Error Code	Module Error Code	Error Message	Description

4440	10100	queue name is not existed, or deleted	The queue does not exist, or the queue has been deleted.
4000	10450	secret id doesn't begin with AKID account	The secret id doesn't begin with AKID account.

Topic Common Error Codes

Error Code	Module Error Code	Error Message	Description
4000	10560	subscription is not existed	Subscription does not exist.
4440	10600	topic name is not existed, or deleted	The topic does not exist, or the topic has been deleted.

Signature Method

Last updated：2018-10-11 20:46:03

Tencent Cloud API will authenticate each access request, so each request is required to include the signature information in the common request parameter for user authentication. The Signature is generated with the user's security credential, which consists of a SecretId and a SecretKey. Users who have no security credential can apply for a credential on the Tencent Cloud. Otherwise, the Cloud API cannot be called.

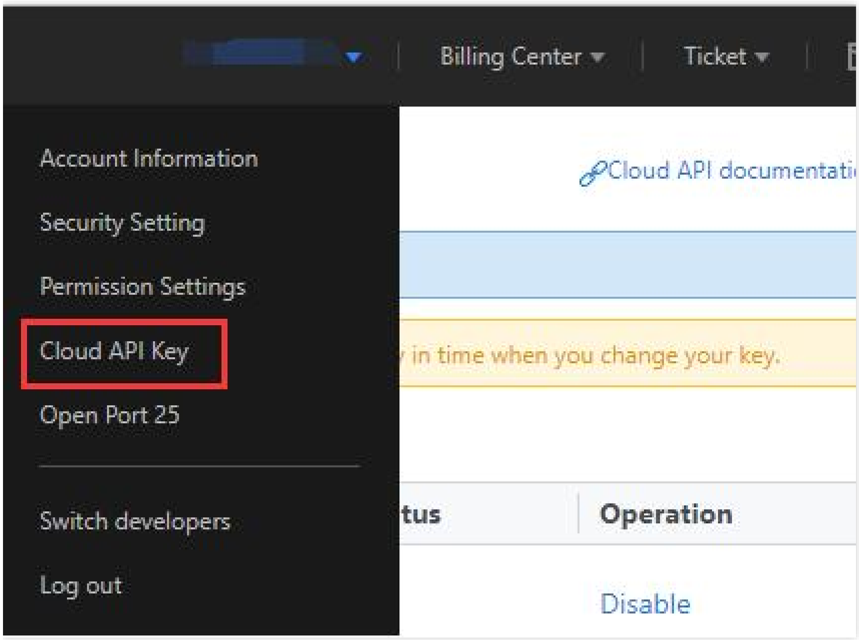
1. Applying for Security Credential

Before using the Cloud API for the first time, a user needs to apply for a security credential on the Tencent Cloud CVM console. A security credential consists of a SecretId, which identifies the API caller, and a SecretKey, which is used to encrypt the signature string and verify the signature string on the server. Users must strictly keep their SecretKeys confidential to avoid disclosure.

To apply for a security credential, please proceed as follows:

1) Log in to the Tencent Cloud Console.

2) Select account name in the top right corner on the navigation bar, and choose "Cloud API Key" in the drop-down box to access the Cloud API key management page.



3) On the Cloud API Key Management page, click "New" to create a pair of SecretId/SecretKey. Each account can have two pairs of SecretId/SecretKey at most.

2. Generating Signature String

With the SecretID and SecretKey, a signature string can be generated. The following is the detailed process for generating a signature string.

If a user has the following SecretId and SecretKey:

```
SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA
SecretKey: Gu5t9xGARNpq86cd98joQYCN3Cozk1qA
```

Note: This is just an example. Please proceed with your actual SecretId and SecretKey!
Take Query Instance List (DescribeInstances) as an example. The possible request parameters are as follows when this API is called:

Parameter name	Description	Parameter Value
Action	Method name	DescribeInstances
SecretId	Key ID	AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA
Timestamp	Current timestamp	1465185768
Nonce	Random positive integer	11886
Region	Indicate the region where the instance is located	gz
instanceIds.0	ID of the instance to be queried	ins-09dx96dg
offset	Offset value	0
limit	Maximum number of output values	20

According to the above table, among the request parameters, there are only 5 common request parameters (Action, SecretId, Timestamp and Nonce), instead of 6 ones as described in "Common Request

Parameters". Actually, Region is not mandatory for CDN, and Signature (the sixth one) is generated from other parameters (including the instruction request parameters) using the following procedure:

2.1. Sorting Parameters

First, sort all request parameters in ascending lexicographical order by their names, just like sorting words in a dictionary in ascending alphabetical order or numerical order. That is to say, sort the parameters by their first letters, and then sort the parameters with the same first letter by their second letters, and so on. You can complete the sorting with the relevant sorting functions in programming language, such as the ksort function in PHP. The sorting result of the above sample parameters is as follows:

```
{
'Action' : 'DescribeInstances',
'Nonce' : 11886,
'Region' : 'gz',
'SecretId' : 'AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA',
'Timestamp' : 1465185768,
'instanceIds.0' : 'ins-09dx96dg',
'limit' : 20,
'offset' : 0,
}
```

Any other programming language can be used to sort these parameters as long as the same result is produced.

2.2. Generating Request String

This step is used to generate a request string.

Format the above sorted parameters as "parameter name=parameter value". Take the parameter "Action" as an example. If the parameter value is "DescribeInstances", the resulting format will be "Action=DescribeInstances".

Note: 1. "Parameter value" is the original value instead of url encoded value. 2. If the input parameter contains an underscore"_", you need to convert it to ".".

Then, joint the formatted parameters together using "&" to generate the final request string:

```
Action=DescribeInstances&Nonce=11886&Region=gz&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx
3gnPhESA&Timestamp=1465185768&instanceIds.0=ins-09dx96dg&limit=20&offset=0
```

2.3. Generating Original Signature String

This step is used to generate an original signature string.

The original signature string is composed of the following parameters:

1) Request method: POST and GET methods are supported. In this case, we use a GET request. Note that the method must be in uppercase.
2) Request CVM: The request domain for View List of Instances (DescribeInstances) is cvm.api.qcloud.com. The actual request domain varies depending on the module to which the API belongs. For more information, refer to the descriptions of each API.
3) Request path: The request path of Cloud API is always /v2/index.php.
4) Request string: This is the request string generated in the previous step.

Combination rule of original signature string:

```
Request method + Request CVM +Request path + ? + Request string
```

The combination result is as follows:

```
GETcvm.api.qcloud.com/v2/index.php?Action=DescribeInstances&Nonce=11886&Region=gz&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA&Timestamp=1465185768&instanceIds.0=ins-09dx96dg&limit=20&offset=0
```

2.4. Generating Signature String

This step is used to generate a signature string.
Sign the**original signature string**obtained in the previous step using HMAC-SHA1 algorithm, and then encode the signature string using Base64 to obtain the final signature string.

For example, the codes are as follows if written in PHP:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3Cozk1qA';
$srcStr = 'GETcvm.api.qcloud.com/v2/index.php?Action=DescribeInstances&Nonce=11886&Region=gz&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA&Timestamp=1465185768&instanceIds.0=ins-09dx96dg&limit=20&offset=0';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

The final signature string is as follows:

```
NSI3UqqD99b/UJb4tbG/xZpRW64=
```

When another programming language is used, you can perform the signature verification using the original signature string in the above example as long as the resulting signature string is identical to the one in the example.

3. Encoding Signature String

The generated signature string cannot be directly used as a request parameter, and needs to be encoded with URL encoding.

Note: If the GET method is used, all request parameters need to be encoded with URL encoding.

For example, the signature string generated in the previous step is: NSI3UqqD99b/UJb4tbG/xZpRW64=. When encoded, it should be: NSI3UqqD99b/UJb4tbG/xZpRW64=. The resulting signature string request parameter (Signature) is NSI3UqqD99b/UJb4tbG/xZpRW64=, which will be used to generate the final request URL.

Queue Model

Queue APIs

Creating a Queue

Last updated：2018-06-22 17:19:18

1. API Description

This API (CreateQueue) is used to create a new queue under the user's account.

Domain for public network API request: cmq-queue-region.api.qcloud.com

Domain for private network API request: cmq-queue-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. Hence it is strongly recommended to use **private network** domains, in which traffic consumption is NOT charged.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support HTTP and HTTPS. Private network requests only support HTTP.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.
maxMsgHeapNum	No	Int	Maximum number of messages in the queue. The value range is `1,000,000 - 10,000,000` during beta test, and will be increased to `1,000,000 - 1,000,000,000` when officially launched. The default value is `10,000,000` during beta test, and `100,000,000` when officially launched.
pollingWaitSeconds	No	Int	Waiting time for messages to be received when using long-polling. The value ranges from 0 to 30 seconds. Default is 0.
visibilityTimeout	No	Int	Message visibility timeout. Value range is 1-43200 seconds (within 12 hours). Default is 30.
maxMsgSize	No	Int	Maximum message length. Value range is 1024-65536 Bytes (1-64 K). Default is 65536.
msgRetentionSeconds	No	Int	Message retention period. Value range is 60-1296000 seconds (1 min-15 days). Default value is 345600 (4 days).
rewindSeconds	No	Int	Indicate whether the feature of message rewind is enabled for the queue. Value range is 0-msgRetentionSeconds, where 0 means not enabling the feature, and msgRetentionSeconds, which refers to the period for which the message is retained in the queue, is the maximum rewind time.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.

Parameter Name	Type	Description
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
queueId	String	Unique ID of the queue.

Error Code	**Module Error Code**	**Error Message**	**Description**
4000	10020	Queue name format error	The format of the queue name is incorrect. See the description of the queueName field for the API CreateQueue for the rules.
4450	10220	Number of queues has reached the limit	The number of queues has reached the upper limit. The number of queues for an account is limited (1,000). When the maximum number is reached, no queue can be created then.
6040	10660	It will take some time to release resources of previous queue before you create a new one with the same name, please try later	Failed to create a queue with the same name. It will take some time to release resources of the previous queue with the same name. Currently, to ensure data consistency of CMQ, it is not allowed to create a new queue with the same name within ten seconds after a queue is deleted.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=CreateQueue
&queueName=test-queue-123
&<Common Request Parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"queueId":"queue-ajksdfasdowe"
}
```

Getting Queue List

Last updated：2018-09-14 13:30:29

1. API Description

This API (ListQueue) is used to display the queue list under the user's account, and the list can be obtained by page.

Domain for public network API request: cmq-queue-region.api.qcloud.com

Domain for private network API request: cmq-queue-region.api.tencentyun.com

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support HTTP.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
searchWord	No	String	The parameter is used to filter queue lists. The backend will return queue lists which meet the conditions through fuzzy match. If this parameter is not specified, all the queues under the account will be returned by default.
offset	No	Int	The starting position from which the current page get the queue list when paging query is performed. If this value is specified, the "limit" is also required. By default, the backend will take the value as 0

Parameter Name	Required	Type	Description
limit	No	Int	Number of queues obtained in this page when paging query is performed. If the parameter is not passed, the default value of 20 will be taken.

3. Output Parameters

Parameter Name	Type	Description
code	Int	Please refer to Error Codes
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
totalCount	Int	It indicates the total number of queues returned under the user account for this request, not the number of queues acquired in this page after paging query is performed.
queueList	Array	Queue list information. Each element indicates the information of one queue.

queueList is defined as follows:

Parameter Name	Type	Description
queueId	String	Unique ID of the queue.
queueName	String	Queue name. This is unique under the same account in one region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.

4. Example

Input:

```
https://domain/v2/index.php?Action=ListQueue
&searchWord=test
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"totalCount":10
"queueList":
[
{
"queueId":"queue-asdfo",
"queueName":"test-queue1"
}
,
{
"queueId":"queue-asdsafo",
"queueName":"queue-test1"
}
]
}
```

Getting Queue Properties

Last updated：2018-06-22 17:18:51

1. API Description

This API (GetQueueAttributes) is used to acquire the attributes of a created queue. Apart from the configurable attributes that were configured when creating the queue, the returned attributes will also include the creation time of the queue, the last modification time of the queue and the statistical information about the messages in the queue (approximate value).

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

- region should be replaced by specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The region value in the common parameter should be kept consistent with the one of the domain. In case of inconsistency, the domain region should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeed, 4440: Queue does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
maxMsgHeapNum	Int	Maximum number of messages in the queue. The available value range during beta test is `1,000,000 - 10,000,000`. It will be increased to `1,000,000-1,000,000,000` when officially launched. The default value is `10,000,000` during beta test, and `100,000,000` when officially launched.
pollingWaitSeconds	Int	Waiting time for messages to be received when using long-polling. Value range is 0-30 seconds. Default is 0.
visibilityTimeout	Int	Message visibility timeout. Value range is 1-43200 seconds (within 12 hours). Default is 30.
maxMsgSize	Int	Maximum message length. Value range is 1024-65536 Byte (1-64 K). Default is 65536.
msgRetentionSeconds	Int	How long the messages will be kept. Value range is 60-1296000 seconds (1 min-15 days). Default value is 345600 (4 days).
createTime	Int	Queue creation time. A Unix timestamp will be returned (accurate to second).
lastModifyTime	Int	The time when the queue attributes were modified for the last time. A Unix timestamp will be returned (accurate to second).
activeMsgNum	Int	Total number of messages in the queue whose status is Active (i.e. not Consumed). This is an approximate value.
inactiveMsgNum	Int	Total number of messages in the queue whose status is Inactive (i.e. being consumed). This is an approximate value.
rewindSeconds	Int	The maximum rewind time for messages in the queue. Value range is 0-43200 seconds. 0 means message rewind is disabled.

Parameter Name	Type	Description
rewindmsgNum	Int	Number of messages that has been deleted by calling the DelMsg API but are still within the rewind time.
minMsgTime	Int	Minimum time for messages to be in the "not consumed" status (in seconds).
delayMsgNum	Int	Number of delayed messages.

4. Example

Input:

```
https://domain/v2/index.php?Action=GetQueueAttributes
&queueName=test-queue-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"maxMsgHeapNum": 10000000,
"pollingWaitSeconds": 10,
"visibilityTimeout": 0,
"maxMsgSize": 65536,
"msgRetentionSeconds": 1296000,
"createTime":1462268960,
"lastModifyTime": 1462269960,
"activeMsgNum": 10000,
"inactiveMsgNum": 1000
}
```

Modifying Queue Properties

Last updated：2018-06-22 17:17:00

1. API Description

This API (SetQueueAttributes) is used to modify the attributes of a message queue. Note that you cannot modify the name of a queue.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

- region should be replaced by specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The region value in the common parameter should be kept consistent with the one of the domain. In case of inconsistency, the domain region should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

Parameter Name	Required	Type	Description
maxMsgHeapNum	No	Int	Maximum number of messages in the queue. The available value range during beta test is 1,000,000 - 10,000,000 . It will be increased to 1,000,000-1,000,000,000 when officially launched. The default value is 10,000,000 during beta test, and 100,000,000 when officially launched.
pollingWaitSeconds	No	Int	Waiting time for messages to be received when using long-polling. Value range is 0-30 seconds. Default is 0.
visibilityTimeout	No	Int	Message visibility timeout. Value range is 1-43200 seconds (within 12 hours). Default is 30.
maxMsgSize	No	Int	Maximum message length. Value range is 1024-65536 Byte (1-64 K). Default is 65536.
msgRetentionSeconds	No	Int	How long the messages will be kept. Value range is 60-1296000 seconds (1 min-15 days). Default value is 345600 (4 days).
rewindSeconds	No	Int	Maximum rewind time for messages. Value range is 0-msgRetentionSeconds. The max value equals to the time for which messages will be kept in the queue. 0 means message rewind is disabled.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeed, 4440: Queue does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
maxMsgHeapNum	Int	(Modified) Maximum number of messages in the queue.

Parameter Name	Type	Description
pollingWaitSeconds	Int	(Modified) Waiting time for messages to be received when using long-polling.
visibilityTimeout	Int	(Modified) Message visibility timeout.
maxMsgSize	Int	(Modified) Maximum message length.
msgRetentionSeconds	Int	(Modified) How long the messages will be kept.
rewindSeconds	Int	(Modified) Maximum rewind time for messages.

4. Example

Input:

```
https://domain/v2/index.php?Action=SetQueueAttributes
&queueName=test-queue-123
&pollingWaitSeconds=20
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"queueId":"queue-ajksdfasdowe",
"maxMsgHeapNum":10000000,
"pollingWaitSeconds":20,
"visibilityTimeout":0,
"maxMsgSize":65536,
"msgRetentionSeconds":345600
}
```

Deleting Queues

Last updated：2018-05-15 11:11:03

1. API Description

This API (DeleteQueue) is used to delete a created queue.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

- region should be replaced by specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The region value in the common parameter should be kept consistent with the one of the domain. In case of inconsistency, the domain region should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeed, 4440: Queue does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

4. Example

Input:

```
https://domain/v2/index.php?Action=DeleteQueue
&queueName=test-queue-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Backtracking queue

Last updated：2018-05-15 11:11:41

1. API Description

This API (RewindQueue) is used for message rewind. For more information, please see Message Rewind.
Domain for public network API request: cmq-queue-region.api.qcloud.com

Domain for private network API request: cmq-queue-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The queue name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.

Parameter Name	Required	Type	Description
startConsumeTime	Yes	Unix timestamp	If this time is set, API (Batch) receiveMessage will consume messages after this timestamp based on the order of production messages.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.

Error Code	**Module Error Code**	**Error Message**	**Description**
6050	10670	your queue cannot be rewinded	Queue cannot be rewound.
4000	10680	invalid rewind time, you can get the vaild range of that via GetQueueAttributes	Invalid rewind time. You can view the queue attributes to get the correct rewind time.
4000	10700	rewindSeconds cannot be greater than msgRetentionSeconds	The rewind time cannot be greater than the message retention period.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=RewindQueue
&queueName=test-queue-123&startConsumeTime=unix timestamp
```

```
 &<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Message APIs

Sending Messages

Last updated：2018-05-15 11:12:23

1. API Description

This API (SendMessage) is used to send a message to the specified queue.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
msgBody	Yes	String	Message body. At least 1 Byte. Maximum length depends on the maximum length attribute of queue messages you have configured.
delaySeconds	No	Int	Time needed before the message becomes visible to user after it has been sent to the queue (in seconds).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by the server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
msgId	String	Unique ID of the message generated by the server.

Error Code	**Module Error Code**	**Error Message**	**Description**
4400	10230	exceed maximum message size	There's at least one message that reached the maximum message length limit. You can use the GetQueueAttributes API to view the maximum message length for the queue.
4000	10120	message body can't be empty	The message body of msgBody cannot be empty. It should be at least 1 Byte.
4410	10240	reach maximum	The maximum number of messages in a queue has been reached. You can use the GetQueueAttributes API to view the

		retention number of message	maximum number of messages for the queue.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=SendMessage
&queueName=test-queue-123
&msgBody=helloworld
&<Common Request Parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgId":"123345346"
}
```

Sending Messages in Batch

Last updated：2018-05-15 11:13:31

1. API Description

This API (BatchSendMessage) is used to send a batch of messages (currently you can send up to 16) to the specified queue.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
msgBody.n	Yes	String	Message body. One of the messages in the batch. Currently, there can be up to 16 messages in one message batch. To make it more convenient for users, "n" may start from either 0 or 1, but is must be continuous. For example, if you send two messages, they can be (msgBody.0, msgBody.1) or (msgBody.1, msgBody.2). Note: Currently the total size of all messages is limited to within 64 k (only the size of msgBody, message header and other parameters are not included), it is suggested to make a plan about how many messages to send in the batch beforehand.
delaySeconds	No	Int	Time needed before the message becomes visible to user after it has been sent to the queue (in seconds). (This delay is effective to a batch of messages. Many-to-many mapping is not supported)

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by the server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
msgList	Array	List of unique IDs of the messages generated by the server. Each element is the information of one message.

msgList is defined as follows

Parameter Name	Type	Description
msgId	String	Unique ID of the message generated by the server.

Error Code	**Module Error Code**	**Error Message**	**Description**
4400	10230	exceed maximum message size	There's at least one message that reached the maximum message length limit. You can use the GetQueueAttributes API to view the maximum message length for the queue.
4000	10120	message body can't be empty	The message body of msgBody cannot be empty. It should be at least 1 Byte.
4410	10240	reach maximum retention number of message	The maximum number of messages in a queue has been reached. You can use the GetQueueAttributes API to view the maximum number of messages for the queue.
4470	10300	total message size exceed 64k	The total size of all message bodies cannot exceed 64 k. For batch operation APIs, it is possible for the total size of all message bodies (msgBody only) to exceed 64 k. We established this limit in order to reduce network traffic load and ensure a good user experience for users. It is suggested to decrease the number of messages in the batch if you encountered this error.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=BatchSendMessage
&queueName=test-queue-123
&msgBody.1=helloworld1
```

```
&msgBody.2=helloworld2
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgList":
[
{
"msgId":"123345346"
},
{
"msgId":"456436346"
}
]
}
```

Receiving a Message

Last updated：2018-05-15 11:14:02

1. API Description

This API (ReceiveMessage) is used to consume a message in the queue. The ReceiveMessage operation will change the status of the received message into "inactive". The duration of "inactive" status is determined by the queue attribute visibilityTimeout (Refer to CreateQueue API for details). The consumer needs to delete the message using (batch)DeleteMessage API after the message has been successfully consumed within visibilityTimeout. Otherwise the message will return to "active" status and can be consumed by consumers again.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
pollingWaitSeconds	No	Int	Long-polling waiting time for this request. Value range is 0-30 seconds. If not configured, the value of the queue attribute pollingWaitSeconds will be used by default.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
msgBody	String	Body of the message consumed this time.
msgId	String	Unique ID of the message consumed this time.
receiptHandle	String	Unique receipt handle that is returned each time a message is consumed. This is used to delete the message. Only the receipt handle that was generated when consuming the message for the previous time can be used to delete the message.
enqueueTime	Int	Time to wait for a message to enter the queue after it has been created. A Unix timestamp will be returned (accurate to second).
firstDequeueTime	Int	The time when the message was consumed for the first time. A Unix timestamp will be returned (accurate to second).
nextVisibleTime	Int	The time when the message becomes visible again (i.e. can be consume again). A Unix timestamp will be returned (accurate to second).

Parameter Name	Type	Description
dequeueCount	Int	How many times the message has been consumed.

Error Code	**Module Error Code**	**Error Message**	**Description**
7000	10200	no message	No message in the queue. Strictly speaking, this is not considered an error. Users can simply ignore it and continue to receive messages.
6070	10690	too many unacked(inactive messages or delayed messages)	There are too many invisible or delayed messages in the queue, in which case users are advised to wait for a moment before consuming.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=ReceiveMessage
&queueName=test-queue-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgBody":"helloworld1",
"msgId":"123345346",
"receiptHandle": "283748239349283",
"enqueueTime": 1462351990,
"firstDequeueTime": 1462352990,
"nextVisibleTime": 1462352999,
"dequeueCount": 2
}
```

Receiving Messages in Batch

Last updated：2018-05-15 11:14:22

1. API Description

This API (BatchReceiveMessage) is used to consume multiple messages in the queue (currently, you can consume up to 16 at a time). The BatchReceiveMessage operation will change the status of received messages into "inactive". The duration of "inactive" status is determined by the queue attribute visibilityTimeout (Refer to CreateQueue API for details). The consumer needs to delete the message using (batch)DeleteMessage API after the message has been successfully consumed within visibilityTimeout. Otherwise the message will return to "active" status and can be consumed by consumers again.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
numOfMsg	Yes	Int	Number of messages consumed this time. Value range is 1-16.
pollingWaitSeconds	No	Int	Long-polling waiting time for this request. Value range is 0-30 seconds. If not configured, the value of the queue attribute pollingWaitSeconds will be used by default.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
msgInfo	Array	List of messages. Each element is the detailed content of a message.

msgInfo is defined as follows:

Parameter Name	Type	Description
msgBody	String	Body of the consumed message.
msgId	String	Unique ID of the consumed message.
receiptHandle	String	Unique receipt handle that is returned each time a message is consumed. This is used to delete message. Only the handle that was generated when consuming the message for the previous time can be used to delete the message.
enqueueTime	Int	Time to wait for a message to enter the queue after it has been created. A Unix timestamp will be returned (accurate to second).

Parameter Name	Type	Description
firstDequeueTime	Int	The time when the message was consumed for the first time. A Unix timestamp will be returned (accurate to second).
nextVisibleTime	Int	The time when the message becomes visible again (i.e. can be consume again). A Unix timestamp will be returned (accurate to second).
dequeueCount	Int	How many times the message has been consumed.

Error Code	**Module Error Code**	**Error Message**	**Description**
7000	10200	no message	No message in the queue. Strictly speaking, this is not considered an error. Users can simply ignore it and continue to receive messages.
6070	10690	too many unacked(inactive messages or delayed messages)	There are too many invisible or delayed messages in the queue, in which case users are advised to wait for a moment before consuming.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=BatchReceiveMessage
&queueName=test-queue-123
&numOfMsg=2
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgInfoList":
[
{
"msgBody":"helloworld1",
"msgId":"123345346",
"receiptHandle": "283748239349283",
"enqueueTime": 1462351990,
"firstDequeueTime": 1462352990,
"nextVisibleTime": 1462352999,
"dequeueCount": 2
},
{
"msgBody":"helloworld2",
"msgId":"1233453456",
"receiptHandle": "28374345763283",
"enqueueTime": 1462351990,
"firstDequeueTime": 1462352990,
"nextVisibleTime": 1462352999,
"dequeueCount": 2
}
]
}
```

Deleting a Message

Last updated：2018-05-15 11:14:54

1. API Description

This API (DeleteMessage) is used to delete message that has already been consumed. The consumer needs to use the ReceiptHandle (obtained after the last consume operation) as parameter to locate the message to be deleted. You can only execute this operation before NextVisibleTime. Message will return to Active status after NextVisibleTime and ReceiptHandle will become invalid, causing deletion operation to fail. In this case, you will need to re-consume the message and acquire ReceiptHandle again. Under concurrent consuming scenario, if a message is deleted by one of the consumers, the other consumers will no longer be able to obtain the deleted message.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
receiptHandle	Yes	String	Unique receipt handle that is returned when the message is consumed for the previous time. This is used to delete message.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

Error Code	**Module Error Code**	**Error Message**	**Description**
4430	10260	receipt handle is invalid	Receipt handle is invalid. Refer to Delete Messages for the reasons for why handles become invalid.
4000	10470	receiptHandle error	receiptHandle error. receiptHandle is a string

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=DeleteMessage
&queueName=test-queue-123
&receiptHandle=3423452345
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Deleting Messages in Batch

Last updated：2018-05-15 11:15:31

1. API Description

This API (BatchDeleteMessage) is used to delete a batch of messages (currently you can delete up to 16 messages at a time) that have already been consumed. The consumer needs to use the ReceiptHandle (obtained after the last consume operation) as parameter to locate the messages to be deleted. You can only execute this operation before NextVisibleTime. Messages will return to Active status after NextVisibleTime and ReceiptHandle will become invalid, causing deletion operation to fail. In this case, you will need to re-consume the messages and acquire ReceiptHandle again. Under concurrent consuming scenario, if a message is deleted by one of the consumers, the other consumers will no longer be able to obtain the deleted message.

Domain for public network API request:cmq-queue-region.api.qcloud.com

Domain for private network API request:cmq-queue-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
queueName	Yes	String	Queue name. This is unique under the same account in one region. The name of queue is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
receiptHandle.n	Yes	String	Receipt handle returned when consuming message the last time. To make it more convenient for users, "n" may start from either 0 or 1, but is must be continuous. For example, if you delete two messages, they can be (receiptHandle.0,receiptHandle.1) or (receiptHandle.1, receiptHandle.2).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeed, 4420: Maximum qps limit has been reached, 4440: Queue does not exist, 6010: Failed to delete some of the messages, 6020: Failed to delete any of the messages. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
errorList	Array	List of errors regarding failed deletion operations. Each element contains the error and reason for why a message could not be deleted.

errorList is defined as follows:

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.

Parameter Name	Type	Description
message	String	Error message.
receiptHandle	String	Receipt handle of the message that was not successfully deleted.

Error Code	**Module Error Code**	**Error Message**	**Description**
6010	10150	delete message partially failed	Failed to delete some of the messages in the batch. An error message will be provided for each of the messages that was not successfully deleted. Failed deletions are likely caused by invalid receipt handles.
4430	10260	receipt handle is invalid	Receipt handle is invalid. Refer to Delete Messages for the reasons for why handles become invalid.
6020	10290	batch delete message failed	Failed to batch delete messages.
4000	10470	receiptHandle error	receiptHandle error. receiptHandle is a string

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=BatchDeleteMessage
&queueName=test-queue-123
&receiptHandle.1=3423452345
&receiptHandle.1=4364564575
&<Common request parameters>
```

Output:

When all are successfully deleted

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

When part of the messages are not successfully deleted

```
{
"code" : 6010,
"message" : "delete message partially failed",
"requestId":"14534664555",
"errorList":
[
{
"code" : 4430,
"message" : "invalid receiptHandle",
"receiptHandle":"4364564575"
}
]
}
```

Topic Model

Topic APIs

Creating a Topic

Last updated：2018-05-15 11:15:57

1. API Description

This API (CreateTopic) is used to create a new topic under the user's account.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Note: Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests support both http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.
maxMsgSize	No	Int	Maximum message length. Value range is 1024-65536 Bytes (1-64 K). Default is 65536.
filterType	No	Int	It is used to specify a message matching policy for a topic: filterType = 1 or null indicates that filterTag is used to filter messages for all the subscriptions under the topic; filterType = 2 indicates that bindingKey is used for filtering. Note: This parameter cannot be changed once configured.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
topicId	String	Unique ID of a topic. Please note that Cloud APIs are called through name instead of ID.

4. Error Codes

Error Code	**Module Error Code**	**Error Message**	**Description**
4460	10550	topic is already existed	A topic with the same name exists under the same account.
4000	10590	topic name format	The format of the topic name is incorrect.

		error	
4450	10610	number of topics has reached the limit	The number of topics has reached the limit. Currently, the limit is 1,000.
6040	10660	It will take some time to release resources of previous topic before you can create a new topic with the same name. Please try later.	Failed to create a topic with the same name. It will take some time to release resources of the previous topic with the same name. Currently, to ensure data consistency of CMQ, it is not allowed to create a new topic with the same name within ten seconds after a topic is deleted.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

5. Example

Input:

```
https://domain/v2/index.php?Action=CreateTopic
&topicName=test-topic-123
&<Common Request Parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"topicId":"topic-ajksdfasdowe"
}
```

Modifying Topic Properties

Last updated：2018-05-15 11:16:20

1. API Description

This API (SetTopicAttributes) is used to modify the attributes of a message topic. Please note that the topic name cannot be modified

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.
maxMsgSize	No	Int	Maximum message length. Value range is 1024-65536 Bytes (1-64 K). Default is 65536.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded; 4440: Topic does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.

4. Example

Input:

```
https://domain/v2/index.php?Action=SetTopicAttributes
&topicName=test-Topic-123
&maxMsgSize=1024
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Getting Topic List

Last updated：2018-05-15 11:17:21

1. API Description

This API (ListTopic) is used to display the topic list under a user's account, and the list can be obtained by page.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
searchWord	No	String	It is used to filter topic lists. The backend will return topic lists which meet the conditions through fuzzy match. If this parameter is not specified, all the topics under the account will be returned by default.
offset	No	Int	The starting position from which the current page get the queue list when paging query is performed. If this value is specified, the "limit" is also required. By default, the backend will take the value as 0
limit	No	Int	Number of topics obtained in this page when paging query is performed. If the parameter is not passed, the default value of 20 will be taken.

The meanings of offset and limit here are the same as the offset and limit of SQL.

3. Output Parameters

Parameter Name	Type	Description
code	Int	Please refer to Error Codes
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
totalCount	Int	It indicates the total number of topics returned under the user account for this request, not the number of topics acquired in this page after paging query is performed.
topicList	Array	Topic list information. Each element indicates the information of one topic.

topicList is defined as follows:

Parameter Name	Type	Description

Parameter Name	Type	Description
topicId	String	Unique ID of a topic.
topicName	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.

4. Example

Input:

```
https://domain/v2/index.php?Action=ListTopic
&searchWord=test
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"totalCount":10
"topicList":
[
{
"topicId":"topic-asdfo",
"topicName":"test-topic1"
}
,
{
"topicId":"topic-asdsafo",
"topicName":"topic-test1"
}
]
}
```

Getting Topic Properties

Last updated：2018-05-15 11:20:03

1. API Description

This API (GetTopicAttributes) is used to get the attributes of a created topic. In addition to the configurable attributes that are configured when a topic is created, the returned attributes also include the topic creation time (createTime), the time when the topic attributes were last modified (lastModifyTime) and the statistics about the messages in the topic (approximate value).

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded; 4440: Topic does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
msgCount	Int	Current number of messages in the topic (number of retained messages).
maxMsgSize	Int	Maximum message length. Value range is 1024-65536 Bytes (1-64 K). Default is 65536.
msgRetentionSeconds	Int	The maximum available time of the message in the topic (in seconds). A message will be deleted no matter whether or not it is successfully pushed to users after the period of time specified by this parameter. This parameter value is always one day (86,400 seconds) and cannot be modified.
createTime	Int	Topic creation time. A Unix timestamp will be returned (accurate to second).
lastModifyTime	Int	The time when the topic attributes were last modified. A Unix timestamp will be returned (accurate to second).
filterType	Int	Indicate the filtering policy that a user selects when creating a subscription: filterType = 0 indicates that filterTag is used for filtering; filterType = 1 indicates that bindingKey is used for filtering.

4. Example

Input:

```
https://domain/v2/index.php?Action=GetTopicAttributes
&topicName=test-topic-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgCount":234
"maxMsgSize": 65536,
"msgRetentionSeconds": 1296000,
"createTime":1462268960,
"lastModifyTime": 1462269960,
"filterType":0
}
```

Deleting Topics

Last updated：2018-05-15 11:20:56

1. API Description

This API (DeleteTopic) is used to delete a created topic.

Domain for public network API request: cmq-Topic-region.api.qcloud.com

Domain for private network API request: cmq-Topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.

Error Code	**Module Error Code**	**Error Message**	**Description**
4000	10540	there exists subscriptions under this topic, please unsubscribe all of them before DeleteTopic	Before deleting a topic, you must ensure that there are no subscriptions under this topic. This is to prevent mistaken deletions.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=DeleteTopic
&TopicName=test-Topic-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Message APIs
Publishing a Message

Last updated：2018-05-15 11:26:56

1. API Description

This AP (PublishMessage) is used to publish a message to a specified topic.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.
msgBody	Yes	String	Message body. The minimum length is 1 byte, and the maximum length depends on the set maximum length attribute of the topic message.
msgTag.n	No	String	Tag for message filtering. Message tag (used to filter messages). The maximum number of tags is 5, with each containing not more than 16 characters. It is used in combination with the filterTag parameter of API Subscribe, and the rules are: 1) if filterTag is not configured, subscribers can receive all the messages published on Topic no matter whether msgTag is configured; 2) if filterTag array is specified, subscribers can receive the messages published on Topic only when at least one value in filterTag array also exists in the msgTag array (that is, there is an intersection between filterTag and msgTag); 3) if filterTag array is specified, but msgTag is not configured, subscribers cannot receive any messages published on Topic. 3) can be considered as a special case of 2), and there is no intersection between filterTag and msgTag in this case. The rules are designed to allow subscribers to choose at their own discretion.
routingKey	No	String	This field indicates the routing path for sending messages. Its length should be not more than 64 bytes and contain up to 15 ".", i.e. 16 phrases at most. The "routingKey" cannot be specified randomly when messages are sent to the "exchange" of topic type. The above format requirement must be met. A message with routingKey specified by a subscriber will be pushed to all the consumers whose BindingKey matches with routingKey. There are two matching options: 1 * (asterisk) can be a substitute for a word (a sequence of alphabetic string); 2 # (pound sign) can be used to match one or more characters.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
msgId	String	Unique ID of the message generated by the server.

4. Error Codes

Error Code	**Module Error Code**	**Error Message**	**Description**
4000	10490	number of filterTag exceed limit	The number of filterTag exceeds the limit. Currently, the maximum is 5.
6030	10730	no bindingKey or filterTag matches the routingKey or msgTag	No subscription matches the message tag or routingKey for this publication. Therefore, the message published this time will not be delivered to subscribers.
6030	10650	topic has no subscription, please create a subscription before publishing message	There is no subscriber under this topic. Please create a subscriber before publishing a message.
4000	10700	parameters lack of routingKey	The routingKey parameter is missing.
4000	10720	too many msgTag	There are too many msgTag.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

5. Example

Input:

```
https://domain/v2/index.php?Action=PublishMessage
&topicName=test-topic-123
&msgBody=helloworld
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgId":"123345346"
}
```

Publishing Messages in Batch

Last updated：2018-05-15 11:28:03

1. API Description

This API (BatchPublishMessage) is used to publish a batch of messages (a maximum of 16 currently) to a specified topic.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> Whenever (including during alpha test) any public network downstream traffic is generated from the use of a public network domain, a fee will be charged. It is strongly recommended that users whose services are on the Tencent Cloud use **private network** domains, because no fee will be charged for the traffic consumed in the private network.

- region should be replaced with a specific region: gz (Guangzhou), sh (Shanghai), or bj (Beijing). The region value in the common parameters should be consistent with the region value of the domain. If there is an inconsistency, the request will be sent to the region specified by the domain.
- Public network domain requests both support http and https. Private network requests only support http.
- Some of the input parameters are optional. If not specified, the default value will be taken.
- All the output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in one region. The topic name is a string of no more than 64 characters, which can contain letters, numbers, and hyphens (-) and must begin with a letter.
msgBody.n	Yes	String	Message body. One of the messages in the batch. Currently, there can be up to 16 messages in one batch. To make it more convenient for users, "n" may start from either 0 or 1, but must be consecutive. For example, if you send two messages, they can be (msgBody.0, msgBody.1) or (msgBody.1, msgBody.2). Note: Currently, the total size of all messages is limited to 64 k (only the size of msgBody, and message header and other parameters are not included), so we recommend that you determine the number of messages to be sent in the batch beforehand.
msgTag.n	No	String	Tag for message filtering. Message tag (used to filter messages). The maximum number of tags is 5, with each containing not more than 16 characters. It is used in combination with the filterTag parameter of API Subscribe, and the rules are: 1) if filterTag is not configured, subscribers can receive all the messages published on Topic no matter whether msgTag is configured; 2) if filterTag array is specified, subscribers can receive the messages published on Topic only when at least one value in filterTag array also exists in the msgTag array (that is, there is an intersection between filterTag and msgTag); 3) if filterTag array is specified, but msgTag is not configured, subscribers cannot receive any messages published on Topic. 3) can be considered as a special case of 2), and there is no intersection between filterTag and msgTag in this case. The rules are designed to allow subscribers to choose at their own discretion.

Parameter Name	Required	Type	Description
routingKey	No	String	This field indicates the routing path for sending messages. Its length should be not more than 64 bytes and contain up to 15 ".", i.e. 16 phrases at most. The "routingKey" cannot be specified randomly when messages are sent to the "exchange" of topic type. The above format requirement must be met. A message with routingKey specified by a subscriber will be pushed to all the consumers whose BindingKey matches with routingKey. There are two matching options: 1 * (asterisk) can be a substitute for a word (a sequence of alphabetic string); 2 # (pound sign) can be used to match one or more characters.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. See the table below for specific errors.
message	String	Error message.
requestId	String	Request ID generated by server. When there is an internal error on the server, users can submit this ID to the backend to locate the problem.
msgList	Array	List of unique IDs of the messages generated by the server. Each element indicates the information of one message.

msgList is defined as follows:

Parameter Name	Type	Description
msgId	String	Unique ID of the message generated by the server.

Error Code	**Module Error Code**	**Error Message**	**Description**
4000	10490	number of filterTag	The number of filterTag exceeds the limit.

		exceed limit	Currently, the maximum is 5.
6030	10730	no bindingKey or filterTag matches the routingKey or msgTag	No subscription matches the message tag or routingKey for this publication. Therefore, the message published this time will not be delivered to subscribers.
6030	10650	topic has no subscription, please create a subscription before publishing message	There is no subscriber under this topic. Please create a subscriber before publishing a message.
4000	10700	parameters lack of routingKey	The routingKey parameter is missing.
4000	10720	too many msgTag	There are too many msgTag.

Note: The error codes listed in the above table are specific to this API. If the error code you are looking for is not in the table, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=BatchPublishMessage
&topicName=test-topic-123
&msgBody.1=helloworld1
&msgBody.2=helloworld2
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"msgList":
[
{
"msgId":"123345346"
},
{
"msgId":"456436346"
}
]
}
```

Delivering Messages

Last updated：2018-05-15 11:29:29

Queue Endpoint Subscription

CMQ will push the message text for topic publishing to the subscription Queue, so that consumers can read the corresponding message from the Queue.

Http Endpoint Subscription

Delivery Description

CMQ pushes a topic message to the Http Endpoint of the subscription by sending a POST request. Two message formats are available: JSON and SIMPLIFIED.

JSON format: The Body of the HTTP request contains the body and attributes of the message.

SIMPLIFIED format: The Body of the HTTP request is the message body. Information such as msgId will be sent to the subscriber in the Header of the HTTP request.

If a standard 2xx response (e.g. 200) is returned by the subscriber's HTTP server, it means that the request has been successfully delivered; otherwise, it means that the delivery failed, and the retry policy will be triggered. Response timeout will be deemed as a failure by CMQ, and the retry policy will also be triggered. The duration for timeout checking is about 15 seconds.

Header of HTTP Request

Parameter Name	Description
x-cmq-request-id	requestId of the message for current push
x-cmq-message-id	msgId of the message for current push
x-cmq-message-tag	Tag of the message for current push

Body of HTTP Request

- If the format is JSON, the Body of the HTTP request contains the body and attributes of the message.

Parameter Name	Type	Description

Parameter Name	Type	Description
TopicOwner	String	appid of subscribed topic owner
topicName	String	Topic name
subscriptionName	String	Subscription name
msgId	String	Message ID
msgBody	String	Message body
publishTime	Int	Time when the message is published

- If the format is SIMPLIFIED, the Body of the HTTP request is the body of the message published by a publisher.

Response to HTTP Request

A 2xx response is returned if the request is normally processed by the subscriber's HTTP server; other response codes or response timeout are errors, and the retry policy will be triggered.

Request Example

Assume that the HTTP Endpoint of a subscription is `http://test.com/cgi`

JSON format:

```
POST /cgi HTTP/1.1
Host: test.com
Content-Length: 761
Content-Type: text/plain
User-Agent: Qcloud Notification Service Agent
x-cmq-request-id: 2394928734
x-cmq-message-id: 6942316962
x-cmq-message-tag: a, b

{"TopicOwner":100015036,"topicName":"MyTopic","subscriptionName":"mysubscription","msgId":"6942316962","msgBody":"test message","publishTime":11203432}
```

SIMPLIFIED format:

```
POST /cgi HTTP/1.1
Host: test.com
Content-Length: 123
```

```
Content-Type: text/plain
User-Agent: Qcloud Notification Service Agent
x-cmq-request-id: 2394928734
x-cmq-message-id: 6942316962
x-cmq-message-tag: a, b

test message
```

Subscription APIs

Clearing Subscription Tags

Last updated：2018-05-15 11:30:14

1. API Description

This API (ClearSubscriptionFilterTags) is used to clear up subscription tags.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
subscriptionName	Yes	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=ClearSubscriptionFilterTags
&topicName=test-Topic-123
&subscriptionName=test-subscription-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Creating Subscriptions

Last updated：2018-05-15 11:30:44

1. API Description

This API (Subscribe) is used to create a new subscription under one of a user's topics.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

Parameter Name	Required	Type	Description
subscriptionName	Yes	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
protocol	Yes	String	Subscription protocol. Currently, two types of protocols are supported: HTTP, queue. When using HTTP protocol, users need to build web server to receive messages on their own. When using queue, messages will be automatically pushed to the CMQ queue and users can pull the messages concurrently.
endpoint	Yes	String	endpoint used to receive notification. It varies with the protocol used: for HTTP, endpoint must start with "http://", and host can be either domain or IP; for queue, queueName is entered here. Please note that the push service is not able to push messages to private network currently, so the endpoint will not receive pushed messages if you enter private network domain or IP. Currently, the service can push messages to public network and basic network.
notifyStrategy	No	String	Retry policy for the CMQ push server in case of errors when pushing messages to the endpoint. Available values are: 1) BACKOFF_RETRY (backoff retry). Re-push the message at regular intervals. Discard the message after a certain number of reties have been committed, then continue pushing the next message; 2) EXPONENTIAL_DECAY_RETRY (exponential decay retry). The interval between retries is increased exponentially, for example, retry in 1s first, then 2s, 4s, 8s...Since the cycle of a Topic message is one day, the retry operations will last for at most one day before the message is discarded. Default is EXPONENTIAL_DECAY_RETRY.
notifyContentFormat	No	String	Notification content format. Available values are: 1) JSON; 2) SIMPLIFIED, i.e. raw format. If the protocol is queue, the value must be SIMPLIFIED. If the protocol is HTTP, both values are available. Default is JSON.

Parameter Name	Required	Type	Description
filterTag.n	No	String	Message body. Message tag (used to filter message). There can be up to 5 tags, each containing no more than 16 characters. It is used in combination with the msgTag parameter of (Batch)PublishMessage. Rules: 1) If filterTag is not configured, the subscribers will receive all the messages published on Topic no matter whether msgTag is configured or not; 2) If filterTag array is configured with values, the subscribers will receive messages published on Topic only if at lease one of the values in the array exists in the msgTag array as well (that is, filterTag and msgTag intersect with each other); 3) If filterTag array is configured with values but msgTag is not configured, the subscribers will not receive any messages published on Topic. This can be considered a special case of 2), when there is no intersection between filterTag and msgTag. The design concept of the rules is based on the wills of the subscribers.
bindingKey.n	Yes	String Array	There can be up to 5 bindingKeys, each with a length limit of 64 Bytes. This field indicates the filtering policy used for the subscribers to receive messages. Each bindingKey may contain up to 15 ".", i.e. 16 phrases at most.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

Error Code	**Module Error**	**Error Message**	**Description**

	Code		
4490	10470	subscribtion is already existed	A subscription with the same name already exists under the same Topic of the same account.
4500	10480	number of subscription has reached the limit	The number of subscriptions under the same Topic has exceeded the limit. Currently, the limit is 100.
4000	10490	number of filterTag exceed limit	The number of filterTags exceeded the limit. Currently, the limit is 5.
4000	10500	endpoint format error	Incorrect endpoint format. Possible errors include: 1) URL contains space(s); 2) URL does not start with "http://" for http protocol; 3) Invalid URL; 4) protocol does not match with endpoint.
4000	10510	undefined protocol	Undefined protocol. Please check for spelling errors.
4000	10520	undefined notify retry stragety	Undefined push notification retry policy. Please check for spelling errors.
4000	10530	undefined notify content format	Undefined push notification format. Please check for spelling errors.
4510	10570	url connot contain any blank characters	URL cannot contain any blank characters.
4000	10580	subscription name format error	Incorrect subscription name format.
4000	10620	subscription name format error	Incorrect subscription name format.
4000	10630	illegal endpoint	Invalid endpoint.
4000	10640	notifyContentFormat of protocol queue must be SIMPLIFIED	If the protocol field is "queue", notifyContentFormat must be SIMPLIFIED.
6050	10740	too many filterTag or bindingKey	There are too many filterTag or bindingKey, please check the parameter configuration.
4000	10710	parameters lack of bindingKey	bindingKey is missing.
4000	10670	too many filterTag	There are too many filterTag, please check the number of parameters.

4000	10680	too many bindingKey	There are too many bindingKey, please check the number of parameters.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=Subscribe
&topicName=test-topic-123
&subscriptionName=test-subscription-123
&protocol=http
&endpoint=http://your_host/your_path
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Getting Subscription List

Last updated：2018-08-29 21:14:56

1. API Description

This API (ListSubscriptionByTopic) is used to display the subscription list under one of a user's topics. Data can be acquired in pages.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
searchWord	No	String	It is used to filter subscription lists. The backend will return subscription lists that match with the conditions using fuzzy match method. If this parameter is not specified, all the subscriptions of the account will be returned by default.
offset	No	Int	This is the starting point from where the current page will acquire subscription lists when displaying data in pages. If this value is specified, "limit" is also required. The backend will use 0 as default value. Value range: 0-1000.
limit	No	Int	Number of subscriptions to be acquired by the current page when displaying data in pages. Value range: 0-100. If not specified, this parameter is 20 by default.

The meanings of offset and limit here are the same as the offset and limit of SQL.

3. Output Parameters

Parameter Name	Type	Description
code	Int	Please refer to Error Codes
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
totalCount	Int	Total number of topics returned for the current request, under the user account. (This is not the number of topics acquired by the current page when displaying data in pages).
subscriptionList	Array	Topic list information. Each element indicates the information of one topic.

topicList is defined as follows:

Parameter Name	Type	Description
subscriptionId	String	Subscription ID. It is used when you pull monitoring data.
subscriptionName	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
protocol	String	Subscription protocol. Currently, two types of protocols are supported: HTTP, queue. When using HTTP protocol, users need to build web server to receive messages on their own. When using queue, messages will be automatically pushed to the CMQ queue and users can pull the messages concurrently.
endpoint	String	endpoint used to receive notification. It varies with the protocol: for HTTP, endpoint must start with "http://", and host can be either domain or IP; for queue, queueName is entered here.

4. Example

Input:

```
https://domain/v2/index.php?Action=ListSubscriptionByTopic
&searchWord=abc
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"totalCount":2
"topicList":
[
{
"subscriptionId":"subsc-sdkfl",
"subscriptionName":"test-sub2",
"protocol":"http"
"endpoint":"http://testhost/testpath"
}
,
{
"subscriptionId":"subsc-34lasdk",
"subscriptionName":"test-sub1",
"protocol":"queue"
"endpoint":"test-queue-name"
}
]
}
```

Modifying Subscription Properties

Last updated：2018-05-15 11:30:59

1. API Description

This API (SetSubscriptionAttributes) is used to modify the attributes of a subscription.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

Parameter Name	Required	Type	Description
subscriptionName	Yes	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).
protocol	No	String	Subscription protocol. Currently, two types of protocols are supported: HTTP, queue. When using HTTP protocol, users need to build web server to receive messages on their own. When using queue, messages will be automatically pushed to the CMQ queue and users can pull the messages concurrently.
endpoint	No	String	endpoint used to receive notification. It varies with the protocol: for HTTP, endpoint must start with "http://", and host can be either domain or IP; for queue, queueName is entered here.
notifyStrategy	No	String	Retry policy for the CMQ push server in case of errors when pushing messages to the endpoint. Available values are: 1) BACKOFF_RETRY (backoff retry). Re-push the message at regular intervals. Discard the message after a certain number of reties have been committed, then continue pushing the next message; 2) EXPONENTIAL_DECAY_RETRY (exponential decay retry). The interval between retries is increased exponentially, for example, retry in 1s first, then 2s, 4s, 8s...Since the cycle of a Topic message is one day, the retry operations will last for at most one day before the message is discarded. Default is EXPONENTIAL_DECAY_RETRY.
notifyContentFormat	No	String	Notification content format. Available values are: 1) JSON; 2) SIMPLIFIED, i.e. raw format. If the protocol is queue, the value must be SIMPLIFIED. If the protocol is HTTP, both values are available. Default is JSON.

Parameter Name	Required	Type	Description
filterTag.n	No	String	Message body. Message tag (used to filter message). There can be up to 5 tags, each containing no more than 16 characters. It is used in combination with the msgTag parameter of (Batch)PublishMessage. Rules: 1) If filterTag is not configured, the subscribers will receive all the messages published on Topic no matter whether msgTag is configured or not; 2) If filterTag array is configured with values, the subscribers will receive messages published on Topic only if at lease one of the values in the array exists in the msgTag array as well (that is, filterTag and msgTag intersect with each other); 3) If filterTag array is configured with values but msgTag is not configured, the subscribers will not receive any messages published on Topic. This can be considered a special case of 2), when there is no intersection between filterTag and msgTag. The design concept of the rules is based on the wills of the subscribers.
bindingKey.n	Yes	String Array	There can be up to 5 bindingKeys, each with a length limit of 64 Bytes. This field indicates the filtering policy used for the subscribers to receive messages. Each bindingKey may contain up to 15 ".", i.e. 16 phrases at most.

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded, others: Error. For detailed errors, please refer to the table below.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

Error Code	**Module Error Code**	**Error Message**	**Description**
4000	10490	number of filterTag	The number of filterTag exceeded the limit.

		exceed limit	Currently, the limit is 5.
4000	10520	undefined notify retry stragety	Undefined push notification retry policy. Please check for spelling errors.
4000	10530	undefined notify content format	Undefined push notification format. Please check for spelling errors.

Note: The error codes listed in the above table are specific to the API. If the error code you are looking for is not here, you may find it in the Common Error Codes.

4. Example

Input:

```
https://domain/v2/index.php?Action=SetSubscriptionAttributes
&topicName=test-Topic-123
&subscriptionName=test-subscription-123
&filterTag.0=football
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Deleting Subscriptions

Last updated：2018-05-15 11:31:11

1. API Description

This API (Unsubscribe) is used to delete a created subscription.

Domain for public network API request: cmq-Topic-region.api.qcloud.com

Domain for private network API request: cmq-Topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

Parameter Name	Required	Type	Description
subscriptionName	Yes	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded; 4440: Topic does not exist. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.

4. Example

Input:

```
https://domain/v2/index.php?Action=Unsubscribe
&topicName=test-Topic-123
&subscriptionName=test-subscription-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555"
}
```

Geting Subscription Properties

Last updated：2018-05-15 11:31:43

1. API Description

This API (GetSubscriptionAttributes) is used to acquire the attributes of a created subscription.

Domain for public network API request: cmq-topic-region.api.qcloud.com

Domain for private network API request: cmq-topic-region.api.tencentyun.com

> At any time (including alpha test), any downstream traffic generated when using public network domain will incur traffic fee. It is strongly recommended that users on Tencent Cloud use **private network** domain, as it will not incur any traffic fee.

- The "region" should be replaced by a specific region: gz (Guangzhou), sh (Shanghai), bj (Beijing). The "region" value in the common parameter should be kept consistent with that in the domain. In case of inconsistency, the one in the domain should prevail. The request should be sent to the region specified by the domain.
- Requests for accessing via public network domain support both HTTP and HTTPS. Requests for accessing via private network only support HTTP.
- Some of the input parameters are optional, so the default values are not required.
- All output parameters will be returned to the user when the request is successful; otherwise, at least code, message, and requestId will be returned.

2. Input Parameters

The following request parameter list only provides API request parameters. For other parameters, refer to Common Request Parameters.

Parameter Name	Required	Type	Description
topicName	Yes	String	Topic name. It is unique under the same account in a single region. Topic name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

Parameter Name	Required	Type	Description
subscriptionName	Yes	String	Subscription name. It is unique for the same topic under the same account in a single region. Subscription name is a string with no more than 64 characters. It must start with letter, and the rest may contain letters, numbers and dashes (-).

3. Output Parameters

Parameter Name	Type	Description
code	Int	0: Succeeded. For the meanings of other returned values, please refer to Error Codes.
message	String	Error message.
requestId	String	ID of the request generated by server. When there is an internal error on the server, users can submit this ID to backend to locate the problem.
topicOwner	String	Topic owner's appid.
msgCount	Int	Number of messages waiting to be sent for the subscription.
protocol	String	Subscription protocol. Currently, two types of protocols are supported: HTTP, queue. When using HTTP protocol, users need to build web server to receive messages on their own. When using queue, messages will be automatically pushed to the CMQ queue and users can pull the messages concurrently.
endpoint	String	endpoint used to receive notification. It varies with the protocol: for HTTP, endpoint must start with "http://", and host can be either domain or IP; for queue, queueName is entered here.

Parameter Name	Type	Description
notifyStrategy	String	Retry policy for the CMQ push server in case of errors when pushing messages to the endpoint. Available values are: 1) BACKOFF_RETRY (backoff retry). Re-push the message at regular intervals. Discard the message after a certain number of reties have been committed, then continue pushing the next message; 2) EXPONENTIAL_DECAY_RETRY (exponential decay retry). The interval between retries is increased exponentially, for example, retry in 1s first, then 2s, 4s, 8s...Since the cycle of a Topic message is one day, the retry operations will last for at most one day before the message is discarded. Default is EXPONENTIAL_DECAY_RETRY.
notifyContentFormat	String	Notification content format. Available values are: 1) JSON; 2) SIMPLIFIED, i.e. raw format. If the protocol is queue, the value must be SIMPLIFIED. If the protocol is HTTP, both values are available. Default is JSON.
createTime	Int	Subscription creation time. A Unix timestamp will be returned (accurate to second).
lastModifyTime	Int	The time when the subscription attributes were last modified. A Unix timestamp will be returned (accurate to second).
bindingKey	String Array	The filtering policy used for the subscribers to receive messages.

4. Example

Input:

```
https://domain/v2/index.php?Action=GetSubscriptionAttributes
&topicName=test-topic-123
&subscriptionName=test-subscription-123
&<Common request parameters>
```

Output:

```
{
"code" : 0,
"message" : "",
"requestId":"14534664555",
"topicOwner":"1231884",
"msgCount":234,
"protocol":"http",
"endpoint":"http://testhost/testpath",
"notifyStrategy":"EXPONENTIAL_DECAY_RETRY",
"notifyContentFormat":"SIMPLIFIED",
"createTime":1462268960,
"lastModifyTime": 1462269960
}
```