ALIBABA CLOUD

Alibaba Cloud

Alibaba Cloud Message Queue for MQTT

Product Introduction

Document Version: 20220523

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.What is Message Queue for MQTT?

is lightweight messaging middleware provided by Alibaba Cloud for mobile Internet and IoT scenarios. Traditional messaging middleware is generally used between microservices. , which is designed for IoT scenarios, transmits messages between devices and clouds and realizes the Internet of Everything in a real sense. This topic describes the messaging models, benefits, and scenarios of .

Key terms

- *Topic*: the topic of messages. It is a level-1 message type. A producer sends messages to a topic.
- *Producer*: the producer or publisher who produces and sends messages to a topic.
- *Consumer*: the consumer or subscriber who subscribes to and consumes messages from a topic.
- *Message*: the data that is sent by a producer to a topic and finally transmitted to a consumer.
- *Rule*: the resource that implements data exchanges between and other Alibaba Cloud services.

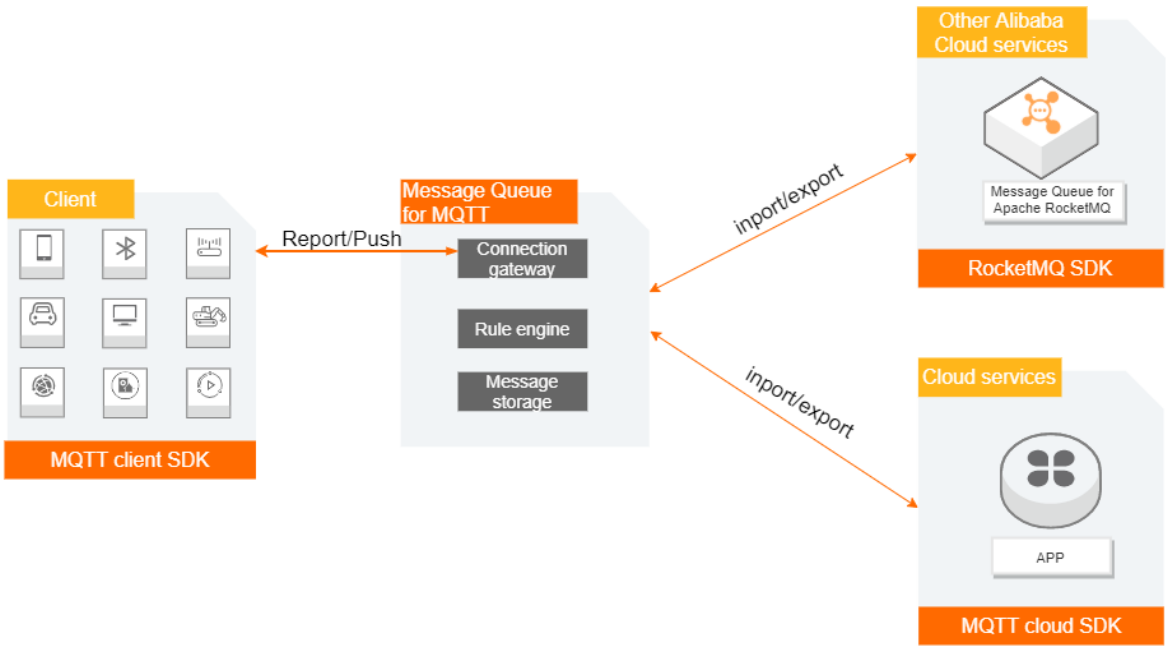
Messaging models

supports the following two messaging models:

- **Model of interaction between devices and backend service applications**

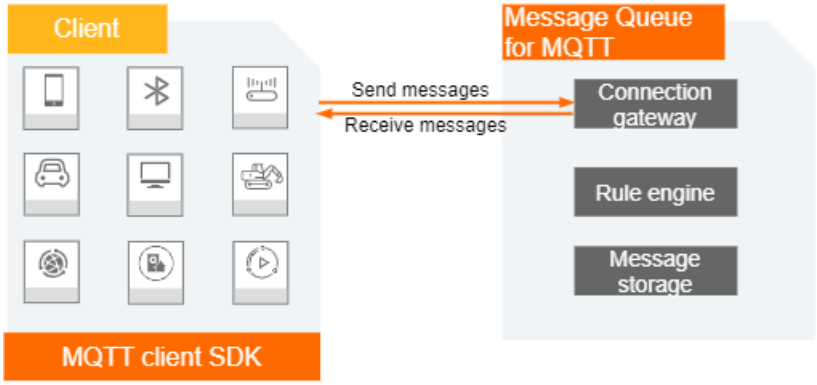
  In this model, connects devices to backend service applications to implement two-way communication between the devices and the backend service applications. Devices can use to directly communicate with backend service applications. Devices can also use Message Queue for MQTT to implement messaging with other Alibaba Cloud services.

  A typical scenario of this model is to report the status data of smart devices to the cloud or control the delivery of commands from backend service applications to smart devices.



- **Model of messaging between devices**

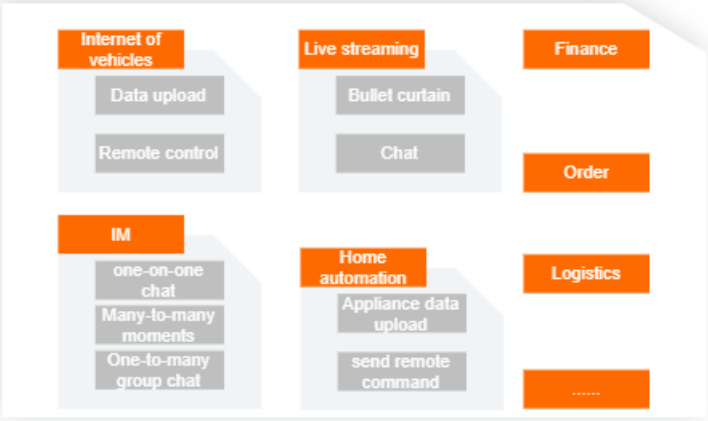
This model is applicable to data communication between mobile applications or devices. Its typical scenarios include chatting between users by using instant messaging technology, and application-based control over smart devices. In this model, message producers and consumers are distributed in the mobile environment and connected to over the Message Queuing Telemetry Transport (MQTT) protocol.



Based on these two models, developers who use are divided into device developers and cloud developers. For more information about the custom development for the developers, see Device developer guide and Cloud developer guide.

Scenarios

supports a range of protocols, languages, and platforms, and is widely used in mobile Internet and IoT fields. This service is applicable to various scenarios, such as mobile live streaming, Internet of Vehicles, finance and payment, smart catering, and instant messaging.



2.Terms

This topic describes the terms of and the Message Queuing Telemetry Transport (MQTT) protocol.

Terms

instance

An entity that you create when you purchase . Each instance maps to a globally unique endpoint. Before you use , you must create an instance in the corresponding region and access the service by using the endpoint of the instance. For more information about how to create a instance, see Use Message Queue for MQTT SDK for Java to send and subscribe to messages without cross-service data exchanges.

message ID

A globally unique identifier for a message. Each message ID is automatically generated by . Message IDs are used to track messages and troubleshoot errors. For more information, see Query the message traces.

Message Queue for MQTT broker

A broker provided by for MQTT-based interactions. A Message Queue for MQTT broker is responsible for messaging between a Message Queue for MQTT client and .

Message Queue for MQTT client

A mobile node that interacts with a broker.

P2P message

A special type of message provided by based on the standard MQTT protocol. This type of message can be sent to a Message Queue for MQTT client without subscriptions. For more information, see P2P messaging for Message Queue for MQTT.

parent topic

MQTT is a messaging protocol based on the publish-subscribe model. Therefore, each message belongs to a topic. The MQTT protocol supports multiple levels of topics. A level-1 topic is a parent topic. Before you use , you must create a parent topic in the console or the console.

subtopic

A level-2 or level-3 topic is a subtopic of a parent topic in Message Queue for MQTT. You can specify subtopics in code without the need to create them in the Message Queue for MQTT console. The naming format is *<Name of the parent topic>/<Name of the level-2 topic>/<Name of the level-3 topic>*. The parent topic and its subtopics are separated by forward slashes (/). Example: SendMessage/demo/producer. The total length of the name for a parent topic and its subtopics cannot exceed 64 characters in . Otherwise, a client exception occurs.

client ID

An identifier that globally and uniquely identifies a client in . If a client uses a client ID that has been used by another client to access , the access request is denied.

A client ID consists of two parts in the format of *<GroupID>*@@@*<DeviceID>*. A client ID cannot exceed 64 characters in length and cannot contain non-printable characters. For more information, see Limits.

group ID

An identifier that specifies a group of nodes with identical logic and features. A group ID represents a set of devices that have the same features. A group ID must be created in the console. For more information about how to create a group ID, see Use Message Queue for MQTT SDK for Java to send and subscribe to messages without cross-service data exchanges.

device ID

An identifier that you specify to uniquely identify each device. Device IDs must be globally unique. For example, you can use the serial number of a sensor as its device ID.

rule

A resource that implements data exchanges between V3.x.x and other Alibaba Cloud services. You can specify the following types of rules in Message Queue for MQTT:

- Data inbound rule: You can configure a data inbound rule to read data from an Alibaba Cloud service and push the data to a Message Queue for MQTT client over the MQTT protocol. This way, you can directly call the API of the specified Alibaba Cloud service to send data to the Message Queue for MQTT client. For more information, see Import data from other cloud services to Message Queue for MQTT.
- Data outbound rule: You can configure a data outbound rule to export messages from a Message Queue for MQTT client to another Alibaba Cloud service. This way, you can directly call the API of the Alibaba Cloud service to read messages sent from the Message Queue for MQTT client. For more information, see Export data from Message Queue for MQTT to other Alibaba Cloud services.
- Rule for client status notification: You can configure a rule for client status notification to export the status event data of a Message Queue for MQTT client to other Alibaba Cloud services. For more information, see Export online and offline events of Message Queue for MQTT clients.

Network-related terms

endpoint

provides both public and internal endpoints. We recommend that you use public endpoints for mobile devices. In addition to standard MQTT port 1883, also supports SSL encryption and WebSocket. The endpoint is automatically allocated after an instance is created. Keep the endpoint for future reference. For more information about how to create an instance, see Use Message Queue for MQTT SDK for Java to send and subscribe to messages without cross-service data exchanges.

MQTT-related terms

MQTT

An industry-standard protocol for the IoT and mobile Internet, which is applicable to data transmission between mobile devices. By default, supports this protocol.

QoS

The quality of service (QoS) level in message transmission, which can be separately set in the producer and consumer.

- The QoS level in the producer affects the transmission quality of messages sent from the producer to .
- The QoS level in the consumer affects the transmission quality of messages sent from the broker to the consumer.

MQTT provides the following QoS levels:

- QoS0: Messages are delivered to intended Message Queue for MQTT clients at most once.
- QoS1: Messages are received by intended Message Queue for MQTT clients at least once.
- QoS2: Messages are delivered to intended Message Queue for MQTT clients exactly once.

cleanSession

In the MQTT protocol, the cleanSession parameter specifies whether a Message Queue for MQTT client in the consumer wants to receive offline messages after a TCP connection is established for it, which is not affected by the configuration in the producer. Set this parameter based on the following syntax:

- cleanSession=true: When an offline Message Queue for MQTT client in the consumer goes online again, all its previous subscriptions and offline messages are cleaned up.
- cleanSession=false: When an offline Message Queue for MQTT client in the consumer goes online again, it processes previous offline messages, and its previous subscriptions remain effective.

Take note of the following points when you use QoS and the cleanSession parameter together:

- In the MQTT protocol, the value of the cleanSession parameter for each client cannot be modified upon each connection. Otherwise, some messages may be mistaken as offline messages.
- In the MQTT protocol, the cleanSession parameter cannot be set to false for messages with QoS2. If a Message Queue for MQTT client subscribes to such messages, the subscription does not take effect even if the cleanSession parameter is set to false.
- The cleanSession parameter of P2P messages is subject to the configuration of the Message Queue for MQTT client that receives the messages.

Combinations of QoS levels and the cleanSession parameter lists the results of different combinations of QoS levels and the cleanSession parameter in the consumer.

Combinations of QoS levels and the cleanSession parameter

QoS level	cleanSession=true	cleanSession=false
QoS0	Offline messages are not delivered. Only one delivery attempt is made for online messages.	Offline messages are not delivered. Only one delivery attempt is made for online messages.
QoS1	Offline messages are not delivered. Online messages are guaranteed to reach the intended Message Queue for MQTT clients.	Offline messages are delivered. Both offline and online messages are guaranteed to reach the intended Message Queue for MQTT clients.

QoS level	cleanSession=true	cleanSession=false
QoS2	Offline messages are not delivered. Online messages are delivered only once.	Not supported

Solution-related terms

RTC
A real-time network communication method for audio and video fields. This method is used in scenarios such as voice calls, video calls, and video conferencing.

RTC server
A server that hosts audio-and-video media channel services, such as related services provided by Alibaba Cloud Real-Time Communication.

audio-and-video service management server
A management node in the RTC system, which is also called an audio-and-video management service. You can develop your own audio-and-video management services to manage the lifecycles of all RTC sessions. Such management nodes are usually deployed on Alibaba Cloud. You can use Alibaba Cloud services to deploy your audio-and-video management services.

mobile audio-and-video application
A terminal application that is used by users in the RTC system. Users use this application to initiate or join a voice or video call.

smart AP
A common network device that supports application programming and can enable Internet access and manage LAN devices. For example, a smart router is a smart access point (AP).

digital price tag
An electronic screen in places such as shopping malls and supermarkets. Digital price tags are networked by using smart AP nodes based on a wireless sensor network protocol such as Bluetooth or ZigBee.

digital price tag management service
The backend service of a digital price tag system. It is used to manage the content that is displayed on the electronic screens and to manage and query manual tasks, such as price changes.

ApsaraDB RDS
A stable, reliable, and scalable online database service provided by Alibaba Cloud. It is used to persistently store task status changes such as price changes in a digital price tag system.

Log Service
A log storage service provided by Alibaba Cloud. This service is used to persistently store all the operation logs in a digital price tag system for auditing and tracing purposes.

3.Limits

This topic describes the limits and specifications for specific metrics in . To prevent exceptions on your applications, set these metrics to valid values based on these limits when you use . The following tables describe the specific limits.

If you use Platinum Edition, you can modify specific metrics based on your business requirements. The following table describes the metrics that you can modify. To obtain technical support and modify metrics, click to submit a ticket.

Limits on instances

Item	Limit	Description
Instance name	• An instance name must be 3 to 64 characters in length. • An instance name can contain digits, letters, hyphens (-), and underscores (_). The name is not case-sensitive. • An instance name must be unique in the same region.	You cannot specify the name of an instance when you create the instance. The default name of a new instance is the ID of the instance. To change the instance name, go to the console.
Number of messaging transactions per second (TPS) on a single instance	Throttling is triggered based on the specifications of the instance that you purchased. The service does not ensure that messages are received by the destination Message Queue for MQTT clients when the specified threshold value in the instance specifications is exceeded.	A single Message Queue for MQTT client can publish and subscribe to only a small number of messages. Specific SDKs do not support a large number of messaging TPS. We recommend that you set the messaging TPS on a single Message Queue for MQTT client to no more than 20. If you require a larger number of messaging TPS on your cloud server, you can use the client.
Number of Message Queue for MQTT clients that are connected to a single instance	Throttling is triggered based on the specifications of the instance that you purchased. The service does not ensure that messages are received by the destination Message Queue for MQTT clients when the specified threshold value in the instance specifications is exceeded.	By default, sends alerts if the maximum number of Message Queue for MQTT clients that are connected to a single instance is exceeded. Change the maximum value to a suitable value.
Number of subscriptions on a single instance	Throttling is triggered based on the specifications of the instance that you purchased. The service does not ensure the integrity of subscriptions when the specified threshold value in the instance specifications is exceeded.	By default, sends alerts if the number of subscriptions on a single instance is exceeded. Change the maximum value to a suitable value.

Item	Limit	Description
IP addresses of the domain name that is used as the endpoint of an instance	No	The IP address may unexpectedly change. Do not assume that the IP address is fixed. The technical team of is not responsible for faults and direct losses or indirect losses in the following scenarios: • Your Message Queue for MQTT client uses an IP address instead of a domain name to access the service. The original IP address becomes invalid after the technical team of Message Queue for MQTT updates the domain name resolution. • A firewall policy on IP addresses is set in the network in which your Message Queue for MQTT client is running. New IP addresses are blocked due to the firewall policy after the technical team of Message Queue for MQTT updates the domain name resolution.

Limits on topics

Item	Limit	Description
Topic name	• A topic name must be 3 to 64 characters in length. **Note** The length of a topic name refers to the total length of the name for a parent topic and the subtopics. • A topic name can contain digits, letters, hyphens (-), and underscores (_). The name is not case-sensitive. • The name of a parent topic on the same instance must be unique.	When you use to publish and subscribe to messages, the length of a topic name must be in the valid range. If the length of the topic is not in the valid range, you cannot publish or subscribe to messages.

Item	Limit	Description
Number of parent topics on a single instance	Maximum value: 25	If the default maximum value cannot meet your requirements, click to submit a ticket to increase the maximum value.
Use of topics across regions	Not supported	A data exchange rule is configured, and relevant resources of and must be in the same region for data exchange. For more information, see Region-specific topics.

Limits on clients

Item	Limit	Description
Client ID	• A client ID must be 0 to 64 characters in length. • A client ID can contain digits, letters, hyphens (-), and underscores (_).	When you use to publish and subscribe to messages, make sure that the client ID does not exceed 64 characters in length. If the client ID exceeds 64 characters in length, the client may be disconnected.
Number of topics to which a single Message Queue for MQTT client can subscribe	30	Each Message Queue for MQTT client can subscribe to a maximum of 30 topics at the same time. If the number of subscribed topics reaches 30, new subscriptions cannot be added. If you use Message Queue for MQTT Platinum Edition, you can click to submit a ticket to increase the maximum value. If topics are subscribed to by using a subscription that contains wildcards, the topics are counted as a subscribed topic. If different subtopics of the same parent topic are subscribed to, each subtopic is counted as a subscribed topic. For example, A/# is counted as a subscribed topic. A/# and A/a1/# are counted as two subscribed topics.

Item	Limit	Description
Group ID	• A group ID must be 7 to 64 characters in length. • A group ID can contain digits, letters, hyphens (-), and underscores (_). It must start with GID_ or GID-. • A group ID must be unique on the same instance.	When you use to publish and subscribe to messages, the length of a group ID must be in the valid range. If the length of the group ID exceeds the valid range, the messages cannot be published or subscribed to.
Number of subscriptions that contain wildcards	Each parent topic supports up to 100 subscriptions that contain wildcards.	The Message Queue for MQTT broker supports a limited number of active subscriptions that contain wildcards for each parent topic. When the maximum value is exceeded, the Message Queue for MQTT broker loads only 100 subscriptions. As a result, some subscribers may fail to receive messages. Therefore, you must control the number of subscriptions that contain wildcards. For example, the subscriptions to A/#, A/a1, and A/a2/# under the same Parent Topic A are counted as three subscriptions.

Limits on rules

Item	Limit	Description
Number of rules for a single instance	100	If the default maximum value cannot meet your requirements, click to submit a ticket to increase the maximum value.
Rule deduplication	Only one rule of the same type can be created for each internal resource.	For example, you can create only one rule for client status notification for each group ID, and one data inbound rule and one data outbound rule for each topic in Message Queue for MQTT.

Limits on publishing and subscribing to messages

Item	Limit	Description

Item	Limit	Description
Message size	64 KB	If the size of a message exceeds the maximum value, the message is discarded. If you use Message Queue for MQTT Platinum Edition, you can click to submit a ticket to increase the maximum value.
Message retention period	Three days	retains offline messages only when the QoS parameter is set to 1 and the cleanSession parameter is set to false. The offline messages can be retained for up to three days and are automatically deleted after this period elapses. For information about the QoS and cleanSession parameters, see Terms. If you use Message Queue for MQTT Platinum Edition, you can click to submit a ticket to increase the message retention period.
QoS and cleanSession configurations	You cannot set the QoS parameter to 2 and the cleanSession parameter to false at the same time.	When you use to publish and subscribe to messages, you can perform the following configurations: • If the cleanSession parameter is set to true, the QoS parameter can be set to 0, 1, or 2. • If the cleanSession parameter is set to false, the QoS parameter cannot be set to 2. does not support this configuration.
Validity period of the token	30 days	When you call the ApplyToken operation, if the ExpireTime parameter is set to a value greater than 30, a token is returned and no error is reported. The validity period of this token is still 30 days.

Item	Limit	Description
Period of waiting for a message timeout or failure	Maximum period: 10 seconds	The first time that the Message Queue for MQTT broker pushes a message, the broker cannot determine whether this message is converted to an offline message before this message times out or fails. The waiting period ranges from 5 seconds to 10 seconds.
Number of stored offline messages	Maximum value: 1,000,000	The Message Queue for MQTT broker of an instance limits the number of offline messages stored on the instance. If the maximum value is exceeded, the Message Queue for MQTT broker deletes the earliest offline messages that are stored. To prevent an excessive number of offline messages, set the cleanSession parameter to a suitable value when you subscribe to topics. If the default maximum value cannot meet your requirements, click to submit a ticket to increase the maximum value.

Limits on cloud API operations

For information about the maximum number of queries per second (QPS) for the cloud API, see QPS limits.