

消息队列 MQTT 版

产品简介



腾讯云

【 版权声明 】

©2013–2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

产品简介

产品概述

应用场景

技术架构

开源对比

使用限制

基本概念

产品简介

产品概述

最近更新时间：2024-12-27 14:25:13

消息队列 MQTT 版（TDMQ for MQTT，简称 TDMQ MQTT 版）是一款分布式高可用的消息队列服务，通过“发布/订阅”模式构建了极其轻量化的消息传输协议。TDMQ MQTT 版兼容标准的 MQTT 协议，支持开源社区和业内主流 MQTT 客户端零改造接入，并在稳定性、低延迟和高性能等特性上做了加强。

产品优势

兼容性

兼容业内标准的 MQTT 3.1.1 协议（未来将继续兼容 MQTT 5.0 协议和更多业内物联网和车联网通用标准协议），使用通用的消息队列的发布和订阅模式，支持一对多的消息发布；支持使用各类开源社区 Java、C/C++、Go 等多语言客户端接入。

可靠性

提供有序、可靠的双向连接，支持 TCP 和 WebSocket 协议。面向受限设备和低带宽、高延迟或不可靠的网络连接场景，做了专门加强和设计。支持 X.509 证书双向认证和“一机一证”等高级特性，也支持用户集成自定义的鉴权系统，连接更加安全可控。

可观测性

控制台提供丰富的监控指标，对接腾讯云的监控和告警功能，同时提供完善的云 API，支持集成自助运维系统。

数据互通

支持和其他产品，如消息队列 RocketMQ、CKafka 和云函数的数据互通，从而实现消息数据的上下游生态和自由流转。

应用场景

最近更新时间：2024-12-27 14:25:13

车联网

车辆通过搭载的传感器和控制器收集各种信息，如车辆位置、速度、油量、发动机状态等，通过 MQTT 协议实时或定期传输到车厂的服务器，以便进行数据分析、故障预警、远程控制等操作。车厂通过 MQTT 协议进行远程命令控制下发，如导航地图更新，修复系统漏洞，远程故障排查等操作。

工业物联网

在工业物联网应用中，各种工业设备需要实时上报数据，并接收远程指令进行操作。MQTT 协议可以作为设备和工业云平台之间的通信协议，实现实时数据上报和远程指令下发。MQTT 可以在弱网环境和极端环境中保持消息的稳定收发，保证了工业场景下的消息有效传输。

设备数据处理平台

由于 MQTT 协议非常适用在低带宽和不稳定的网络环境中，因此非常适合大型数据平台，如智慧城市，城市大脑等物联网数据收集。不同终端，如传感器，移动医用，物联网设备等可以收集环境信息和业务数据传输到下游的数据系统进行处理或者存储。

IM 通信

MQTT 协议采用的发布/订阅模型非常适合即时通信场景，用户可以将消息发布到特定主题，而其他用户只需订阅感兴趣的主题即可接收相关消息。这种模型非常适合实现一对一单聊、多对多群聊等功能。

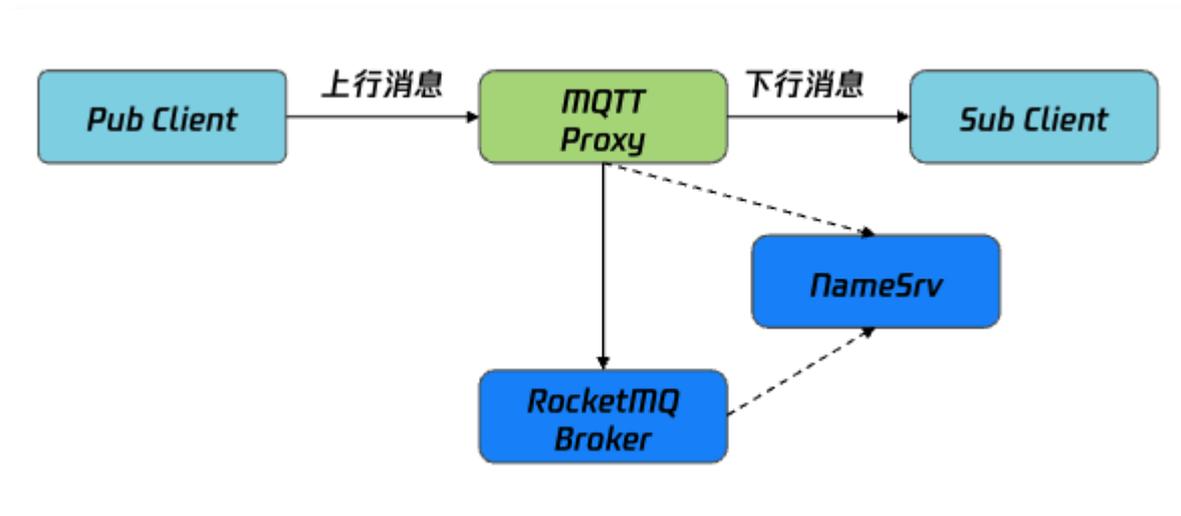
智能家居

在智能家居应用中，各种电器和智能设备需要实时上报数据、接收远程指令进行操作，以及与其他设备进行协同工作。MQTT 协议可以作为智能家居设备和云平台、手机 App 或其他设备之间的通信协议，实现实时数据上报、远程控制和设备协同。

技术架构

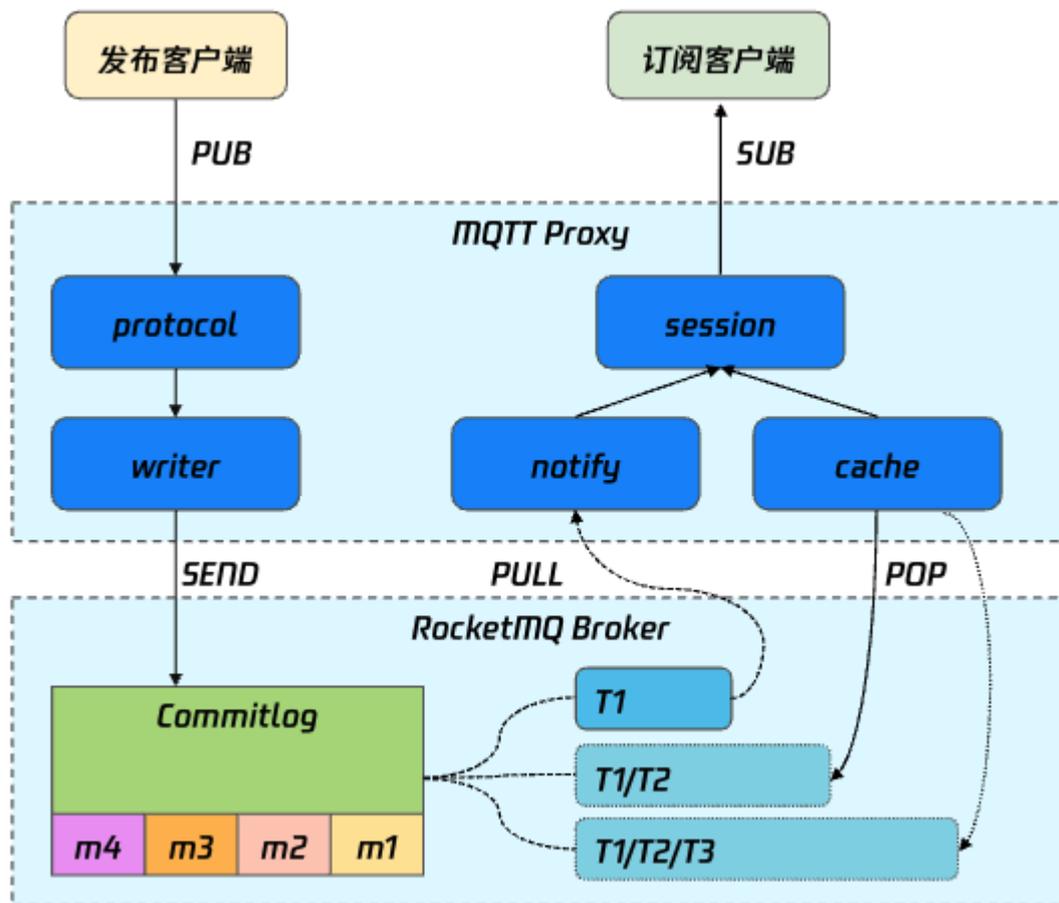
最近更新时间：2024-12-27 14:25:13

MQTT 组件与部署架构



- Pub Client：表示客户端应用或设备，通过 Client ID 唯一标识，可以发布或订阅消息。
- MQTT Proxy：服务端计算组件，负责 MQTT 协议处理，连接会话管理，认证鉴权，无状态可以快速扩缩容。
- NameSrv：命名发现服务，负责按 Topic 路由寻址和 Broker 心跳注册。
- RocketMQ Broker：服务端存储消息的组件，负责消息收发，存储，消息模型，可以横向扩展。

MQTT 实现原理与架构



发布消息流程

发布客户端向 Proxy 建立连接，成功后发布 MQTT 消息，Proxy 将消息解析转换后写入到 RocketMQ Broker 中持久化。

消息存储

Broker 接收消息以后，将消息首先持久到 Commitlog 数据文件中，再根据主题分别建立两个索引，一级主题索引用来做事件通知，多级主题索引用户维护消费进度。

订阅消息流程

订阅客户端向 Proxy 建立连接并指定订阅的主题表达式，Proxy 会如果 session 维护每个连接，当有新消息到达时，通过 notify 模块，广播通知匹配的 session 立即 pop 消息，并且通过 cache 模块，减少重复提取，提升性能和消息实时性。

开源对比

最近更新时间：2024-12-27 14:25:13

TDMQ MQTT 版与开源原生 MQTT 协议的功能对比详情如下：

功能大类	功能细项	TDMQ MQTT 版	其他开源 MQTT 协议实现
基础功能	可视化控制台	具备 MQTT 集群，一级 Topic 和设备等管理页面；QoS，设备订阅关系等信息的详细展示。	不同的实现良莠不齐，部分项目没有标准控制台，黑屏化管理元数据，客户需要自己完成控制台的开发。
	消息查询	可以根据时间范围和一级 Topic 查询消息，未来将开放更多条件查询。	业内主要实现均不支持。
	消息轨迹	根据消息 ID 或者客户端 ID 查询消息的投递轨迹和历史。	业内主要实现均不支持。
	多协议支持	除了 MQTT 协议外，同时支持 TLS 和 WebSocket 等多个协议。	不支持
运维能力	底层架构感知	仅需要关注集群的配额和 TPS 水位，不用关注底层架构。	需要管理底层机器，关注架构和各类机器参数。
	跨 AZ 高可用部署	支持跨 AZ 高可用部署，避免机房级故障。	需要运维人员自行设计部署方案和参数。
	扩缩容能力	用户按需在控制台进行可视化变配。	运维人员手动黑屏操作，自动化、白屏化程度低。
	证书管理	打通腾讯云 SSL 证书管理，页面化管理证书；证书到期前提醒。	需要用户自行管理证书，时刻注意证书过期时间。
可观测性	资源大盘	核心指标观测、生产消费报表和细粒度监控。	不支持
	报警管理	打通腾讯云云监报告警，支持连接数和延迟等指标告警。	不支持
安全管控	分布式限流	保护集群安全，对于超规格的请求进行限流。	不支持，容易出现集群超负荷运转导致集群无法使用的情况。
	认证策略	支持多种认证策略，如 SSL 证书；专业版额外支持“一机一	仅支持最基本的“用户名+密码”认证。

		证”功能，支持对接第三方的认证源，如 JWT, HTTP 服务等。	
	授权策略	支持多维度（客户端 ID, Topic, IP 等）多种授权策略，支持不同策略的优先级管理。	不支持
	主子账号管理	全面支持腾讯云主子账号，实现 CAM 主子账号及企业间跨账号的授权服务。	不支持

使用限制

最近更新时间：2024-12-27 14:25:13

为了保证集群的性能稳定，腾讯云消息队列 MQTT 版根据不同的售卖规格差异进行一些使用限制，不同售卖规格间的主要功能差异请参考 [产品系列](#)。以下介绍您在使用 MQTT 时可能遇到的不同维度的限制。

限制项	具体限制说明	超限制后的行为	建议
集群 TPS 限制	按照购买页选择的规格进行严格限制。 注意： QoS 和 cleanSession 参数的设置会影响 TPS 的计算，1 TPS 并不完全对等1条消息的生产/发送，详见 计算规格说明 。	超出 TPS 限制后，消息的生产/消费请求会被拒绝，并产生报错。在控制台可以查看相关的限流指标，详见 监控说明 。	在购买时请根据实际情况评估流量，当限流发生时，您可以选择 升级集群规格 来应对突发流量。
集群最大在线客户端连接数量	按照购买页选择的规格进行严格限制。	客户端连接被拒绝，连接请求 (CONNECT) 报错。	
集群订阅关系数量	$30 \times$ 最大在线客户端连接数量	新的客户端订阅被拒绝，订阅请求 (SUBSCRIBE) 报错。	在购买时请根据实际情况评估客户端数量，当在线客户端达到上限时，您可以选择 升级集群规格 。
单个客户端 (client id) 订阅 Topic 数	30	新的客户端订阅被拒绝，订阅请求 (SUBSCRIBE) 报错。	
单集群每秒新建连接数	集群最大在线客户端连接数量的 1/5，如集群的连接数上限为 10000，则每秒允许的新建客户端连接数为 2000。	客户端连接被拒绝，连接请求 (CONNECT) 报错。	
单集群一级 Topic 数量	100，考虑到物联网场景的特性，一级 Topic 仅用于区分业务类型，控制数量便于进行业务分组管理和查询等。	无法新建新的一级 topic。	如果实际需求超过当前规格限制可以通过工单 联系我们 。
消息体大小	64 KB	发送消息时会出现消息大小超限额的报错	
消息保留时间	3 天	超出时间的消息会被删除	

基本概念

最近更新时间：2025-02-10 15:01:02

MQTT

一种物联网和移动互联网领域的行业标准协议，适合移动终端之间的数据传输。

MQTT 服务端

实现 MQTT 协议的服务端，负责保持和 MQTT 客户端的长连接，并且将消息路由到对应的客户端，或将消息转发到其他 MQTT 服务端节点，同时负责 MQTT 离线消息的存储。

MQTT 客户端

使用 MQTT 协议连接到 MQTT 服务端的设备或应用程序。

主题

用来标识消息的分类和结构，Publisher 将消息发布到特定的主题上，Subscriber 订阅特定的主题以接收相关信息。

一级 Topic (Parent Topic)

MQTT 协议基于 Pub/Sub 模型，Topic 通常用来对系统生产的各类消息做一个集中的分类和管理。客户端可以订阅一个或多个 Topic，以便于接收和这些主题相关的消息。同样客户端也可以发布消息到某个 Topic，以便其他订阅该 Topic 的客户端接收到这些消息。

二级 Topic (Sub Topic)

根据标准 MQTT 协议，Topic 存在多级，且拥有动态的特性，中间用 “/” 分隔，定义第一级 Topic 为父级 Topic。使用前，需要在 MQTT 控制台创建父 Topic，二级 Topic 无需创建，直接在代码中设置。

通配符

订阅者可以在订阅的主题中使用通配符来达到一次订阅多个主题的目的，MQTT 支持单层通配符 (+) 和 多层通配符 (#) 两种主题通配符，来满足不同的订阅需要。

- 单层通配符 “+” 可用于主题过滤器的任何层级，它可以在主题过滤器的多个层级中使用，并且可以与多层通配符结合使用。例如，“sport/tennis/+” 匹配 “sport/tennis/player1” 和 “sport/tennis/player2”，但不匹配 “sport/tennis/player1/ranking”。此外，由于单层通配符仅匹配单个层级，“sport/+” 不匹配 “sport”，但匹配 “sport/”。
- “#” 可以匹配主题内的任意数量层级，多层通配符代表父级以及任意数量的子级。例如，如果一个客户端订阅了 “sport/tennis/player1/#”，它将接收到这些主题名称发布的消息：“sport/tennis/player1”，

“sport/tennis/player1/ranking”，“sport/tennis/player1/score/wimbledon”。 “sport/#” 也匹配 topic 为 “sport” 的消息，因为 “#” 包括父层级。

会话

会话是指客户端向服务端发起 MQTT 连接请求开始，到连接中断直到会话过期为止的消息收发序列。MQTT 支持两种不同的会话类型：持久会话和清除会话（Clean Session）。

- 持久会话：当客户端连接到服务端时，如果启用了持久会话，服务端会记录客户端的订阅信息，即使客户端断开连接，服务端也会保存这些订阅信息。当客户端再次连接到服务端时，它会恢复之前的订阅信息，并接收它断开连接时的所有未接收的消息。
- 清除会话：当客户端连接到服务端时，如果启用了清除会话，服务端不会保存客户端的订阅信息。当客户端断开连接时，服务端会删除与该客户端相关的所有信息。

ClientID

Client ID 是每个客户端的唯一标识，用户需要自行根据业务场景拼接以保证全局唯一，使用相同的 ClientID 连接 MQTT 服务端会被拒绝。

QoS

QoS（Quality of Service）指代消息传输的服务质量，每条消息都可以在发送时单独设置 QoS。它包括以下级别：

- QoS = 0 代表最多分发一次，消息可能丢失。
- QoS = 1 代表至少达到一次，消息可以保证到达，但是可能重复。
- QoS = 2 代表仅到达一次，消息保证到达，并且不会重复。

Clean Session

Clean Session 是用来控制会话状态生命周期的标志位：

- Clean Session = true：创建一个新的会话，在客户端断开连接时，会话将自动销毁。
- Clean Session = false：创建一个持久会话，在客户端断开连接后会话仍然保持，直到会话超时注销。

P2P

P2P 是一一对一的消息收发模式，Publisher 已经明确该消息的目标接收者信息，并且该消息只需要被指定的单一客户端消费。发送者发送消息时通过设置符合命名规则的 Topic 指定接收者，接收者无需提前订阅即可消费到该消息。

保留消息

保留消息用于确保新订阅 Topic 的设备能及时收到最新状态的消息，即使消息在设备订阅之前发布。当一个客户端发布一条带“保留”标志（Retain）的消息，代理服务器就会保留此条消息。因此，任何订阅了相应消息的主题都将接收到最后一条保留消息，即使最近消息发布者没有在活跃状态。

遗嘱消息

遗嘱消息用于在客户端异常断开连接时向其他订阅者通知其离线状态或执行一些预定义操作。当客户端异常断开连接时（例如网络故障或客户端崩溃），MQTT代理会自动将该遗嘱消息发布给其他订阅者，使得其他订阅者能够得知客户端的离线状态，或者执行一些预定义的操作，例如更新在线状态等。

共享订阅

共享订阅允许多个客户端共享一个主题的订阅，只有一个客户端接收到发布到该主题的消息。当主客户端离线时，能够无缝切换到其他客户端继续接收消息，以确保高可用性。