

分布式消息服务 RabbitMQ

# 用户指南

文档版本 01  
发布日期 2022-08-12



版权所有 © 华为技术有限公司 2022。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 目录

<b>1 产品简介</b>	<b>1</b>
1.1 什么是分布式消息服务 RabbitMQ	1
1.2 产品优势	1
1.3 典型应用场景	2
1.4 产品规格	4
1.5 Kafka 与 RabbitMQ 的差异	5
1.6 与其他云服务的关系	7
1.7 约束与限制	8
1.8 RabbitMQ 相关概念	8
1.9 权限管理	9
<b>2 权限管理</b>	<b>12</b>
2.1 创建用户并授权使用 DMS for RabbitMQ	12
2.2 DMS for RabbitMQ 自定义策略	13
<b>3 环境准备</b>	<b>15</b>
<b>4 购买实例</b>	<b>16</b>
<b>5 连接实例</b>	<b>19</b>
5.1 连接未开启 SSL 方式的 RabbitMQ 实例	19
5.2 连接已开启 SSL 方式的 RabbitMQ 实例	21
5.3 连接 RabbitMQ 管理地址	23
5.4 开启心跳	24
5.5 查看客户端连接地址	26
<b>6 实例日常操作</b>	<b>28</b>
6.1 查看实例	28
6.2 重启实例	30
6.3 删除实例	31
6.4 修改实例信息	32
6.5 重置实例密码	33
6.6 变更实例规格	34
6.7 设置实例的公网访问	35
6.8 设置实例镜像队列	36
6.9 开启实例插件	38

6.10 管理实例标签.....	39
6.11 删除队列.....	41
<b>7 调整资源配额.....</b>	<b>46</b>
<b>8 监控.....</b>	<b>47</b>
8.1 支持的监控指标.....	47
8.2 设置 RabbitMQ 告警规则.....	51
8.3 查看监控数据.....	53
<b>9 云审计服务支持的关键操作.....</b>	<b>54</b>
9.1 云审计服务支持的 DMS for RabbitMQ 操作列表.....	54
9.2 查看云审计日志.....	55
<b>10 常见问题.....</b>	<b>56</b>
10.1 实例问题.....	56
10.1.1 RabbitMQ 使用的版本是多少? .....	56
10.1.2 RabbitMQ 实例 SSL 连接的协议版本号是多少? .....	56
10.1.3 创建实例时为什么无法查看子网和安全组等信息? .....	56
10.1.4 重启 RabbitMQ 实例时, 若其中一台 RabbitMQ 重启失败, 会如何处理? .....	56
10.1.5 RabbitMQ 集群实例如何均衡分发请求到每个虚拟机? .....	56
10.1.6 RabbitMQ 实例集群内部的队列是否有冗余备份? .....	56
10.1.7 RabbitMQ 实例是否支持持久化, 如何定时备份数据? .....	57
10.1.8 RabbitMQ 实例开启 SSL 后, 证书怎么获取? .....	57
10.1.9 RabbitMQ 实例的 SSL 开关是否支持修改? .....	57
10.1.10 RabbitMQ 实例是否支持扩容? .....	57
10.1.11 RabbitMQ 实例是否支持 MQTT 协议? .....	57
10.1.12 如何清空队列数据? .....	58
10.1.13 RabbitMQ 支持升级 CPU 和内存吗? .....	58
10.1.14 如何关闭 RabbitMQ 的 WebUI? .....	59
10.1.15 实例是否支持修改可用区? .....	59
10.2 连接问题.....	60
10.2.1 如何配置安全组? .....	60
10.2.2 RabbitMQ 客户端连接报错原因分析.....	61
10.2.3 RabbitMQ 实例是否支持公网访问? .....	62
10.2.4 RabbitMQ 是否支持跨 Region 部署? .....	62
10.2.5 RabbitMQ 实例是否支持跨 VPC 和跨子网访问? .....	62
10.2.6 RabbitMQ 实例是否支持不同的子网? .....	62
10.2.7 SSL 方式连接 RabbitMQ 实例失败? .....	62
10.2.8 客户端是否可以通过 DNAT 方式访问 RabbitMQ 实例? .....	62
10.2.9 RabbitMQ 实例的 Web 管理页面无法打开.....	62
10.2.10 客户端是否可以连接同个 RabbitMQ 下多个 vhost? .....	63
10.2.11 为什么 RabbitMQ 集群只有一个连接地址? .....	63
10.3 插件问题.....	64
10.3.1 支持的 RabbitMQ 插件有哪些? .....	64

10.4 消息问题.....	64
10.4.1 RabbitMQ 实例支持延时消息队列? .....	65
10.4.2 消息堆积对业务的影响及解决办法.....	65
10.4.3 消息的最长保留时间是多久? .....	67
10.5 监控告警问题.....	67
10.5.1 云监控无法展示 RabbitMQ 监控数据.....	67
10.5.2 云监控显示通道数一直上升报警有影响吗? .....	67
<b>A 修订记录.....</b>	<b>68</b>

# 1 产品简介

## 1.1 什么是分布式消息服务 RabbitMQ

分布式消息服务RabbitMQ完全兼容开源RabbitMQ，为您提供即开即用、消息特性丰富、灵活路由、高可用、监控和告警等特性，广泛应用于秒杀、流控、系统解耦等场景。

- 即开即用  
分布式消息服务RabbitMQ提供单机和集群的消息实例，拥有丰富内存规格，您可以通过控制台直接下单购买并创建，无需单独准备服务器资源。
- 消息特性丰富  
支持AMQP协议，支持普通消息、广播消息、死信、延迟消息等特性。
- 灵活路由  
在RabbitMQ中，生产者将消息发送到交换器，由交换器将消息路由到队列中。交换器支持direct, topic, headers和fanout四种路由方式，同时支持交换机组和自定义。
- 高可用  
RabbitMQ集群提供镜像队列，可通过镜像在其他节点同步数据，单节点宕机时，仍可通过唯一的访问地址对外提供服务，数据不丢失。
- 监控和告警  
支持对RabbitMQ实例状态进行监控，支持对集群每个代理的内存、CPU、网络流量等进行监控。如果集群或节点状态异常，将触发告警。

## 1.2 产品优势

分布式消息服务RabbitMQ完全兼容开源社区版本，旨在为用户提供便捷高效的消息队列。业务无需改动即可快速迁移上云，为您节省维护和使用成本。

- 一键式部署，免去集群搭建烦恼  
专享实例只需要在实例管理界面选好规格配置，提交订单。后台将自动创建部署完成一整套RabbitMQ实例。
- 兼容开源，业务零改动迁移上云

兼容社区版RabbitMQ的API，具备原生RabbitMQ的所有消息处理特性。

业务系统基于开源的RabbitMQ进行开发，只需加入少量认证安全配置，即可使用分布式消息服务RabbitMQ，做到无缝迁移。

### 📖 说明

RabbitMQ专享实例兼容开源社区RabbitMQ 3.7.17版本。

- 独占式体验

RabbitMQ实例采用物理隔离的方式部署，租户独占RabbitMQ实例，每个RabbitMQ之间互不影响。

- 高性能

单队列性能最高可达10万TPS（默认配置），增加队列可获得更高性能。

- 数据安全

独有的安全加固体系，提供业务操作云端审计，消息存储加密等有效安全措施。在网络通信方面，除了提供SASL认证，还借助虚拟私有云（VPC）和安全组等加强网络访问控制。

- 无忧运维

云服务平台提供一整套完整的监报告警等运维服务，故障自动发现和告警，避免7\*24小时人工值守。RabbitMQ专享实例自动上报相关监控指标，如分区数、主题数、堆积消息数等，并支持配置监控数据发送规则，您可以在第一时间通过短信、邮件等获得业务消息队列的运行使用和负载状态。

- 支持多语言客户端

RabbitMQ是一款基于AMQP协议的开源服务，用于在分布式系统中存储转发消息，服务器端用Erlang语言（支持高并发、分布式以及健壮的容错能力等特点）编写，支持多种语言的客户端，如：Python、Ruby、.NET、Java、JMS、C、PHP、ActionScript、XMPP、STOMP、AJAX等。

## 1.3 典型应用场景

RabbitMQ作为一款热门的消息队列中间件，具备高效可靠的消息异步传递机制，主要用于不同系统间的数据交流和传递，在企业解决方案、金融支付、电信、电子商务、社交、即时通信、视频、物联网、车联网等众多领域都有广泛应用。

### 异步通信

将业务中属于非核心或不重要的流程部分，使用消息异步通知的方式发给目标系统，这样主业务流程无需同步等待其他系统的处理结果，从而达到系统快速响应的目的。

如网站的用户注册场景，在用户注册成功后，还需要发送注册邮件与注册短信，这两个流程使用RabbitMQ消息服务通知邮件发送系统与短信发送系统，从而提升注册流程的响应速度。

图 1-1 串行发送注册邮件与短信流程

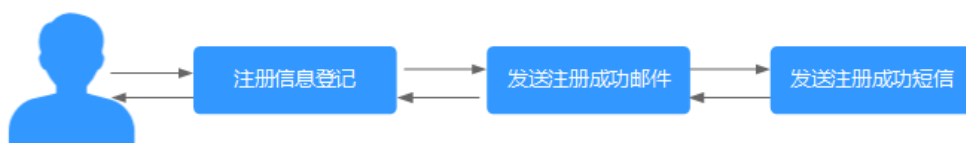
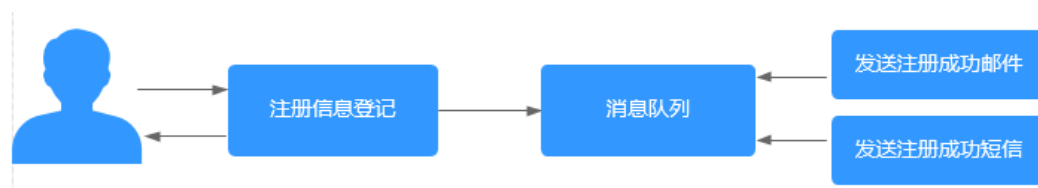


图 1-2 借助消息队列异步发送注册邮件与短信流程

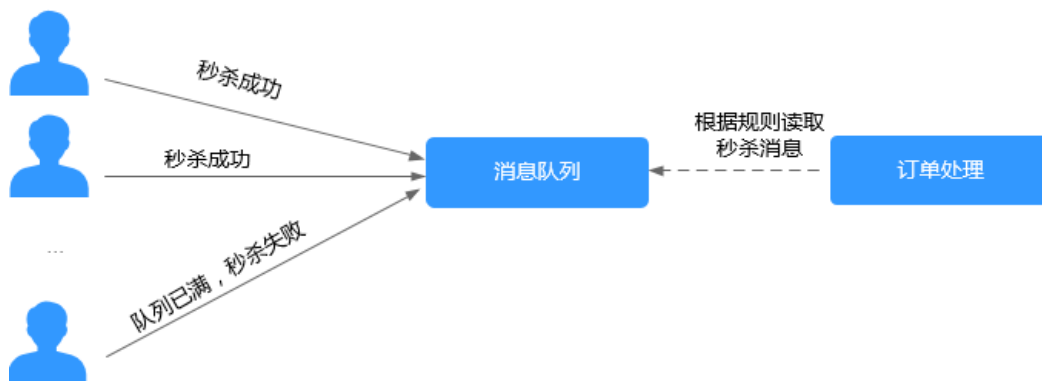


## 错峰流控与流量削峰

在电子商务系统或大型网站中，上下游系统处理能力存在差异，处理能力高的上游系统的突发流量可能会对处理能力低的某些下游系统造成冲击，需要提高系统的可用性的同时降低系统实现的复杂性。电商大促销等流量洪流突然来袭时，可以通过队列服务堆积缓存订单等信息，在下游系统有能力处理消息的时候再处理，避免下游订阅系统因突发流量崩溃。消息队列提供亿级消息堆积能力，3天的默认保留时长，消息消费系统可以错峰进行消息处理。

另外，在商品秒杀、抢购等流量短时间内暴增场景中，为了防止后端应用被压垮，可在前后端系统间使用RabbitMQ消息队列传递请求。

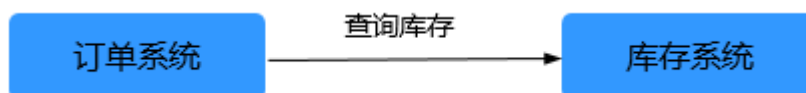
图 1-3 消息队列应对秒杀大流量场景



## 系统解耦

以电商秒杀、抢购等流量短时间内暴增场景为例，传统做法是，用户下单后，订单系统发送查询请求到库存系统，等待库存系统返回请求结果给订单系统。如果库存系统发生故障，订单系统获取不到数据，订单失败。这种情况下，订单系统和库存系统两个子系统高耦合。

图 1-4 系统高耦合

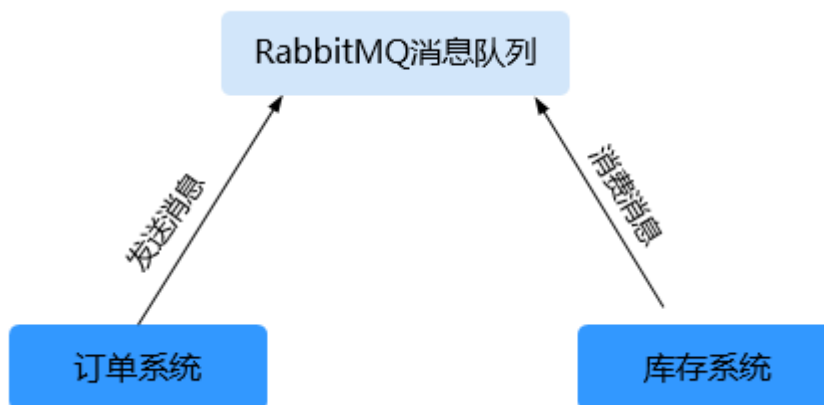


引入RabbitMQ消息队列，当用户下单后，将消息写入到RabbitMQ消息队列中，然后返回用户下单成功。

库存系统订阅下单的消息，消费下单消息，然后进行库操作。即使库存系统出现故障，也不影响用户下单。



图 1-5 系统解耦



## 高可用

镜像队列是开源RabbitMQ 2.6.0版本新增的一个功能，允许集群将队列镜像到其他代理上，当集群某一代理宕机后，队列能自动切换到镜像中的其他代理，保证服务的可用性。

普通队列，由于队列以及队列内容仅存储在单代理上，当该代理故障后，对应的队列不可用。

RabbitMQ引入镜像队列机制，将队列镜像到集群中的其他代理上，每一个镜像队列包含一个主队列和多个从队列，并分布在集群的不同代理上。

## 1.4 产品规格

### RabbitMQ 专享实例规格

RabbitMQ专享版兼容开源RabbitMQ 3.7.17，实例类型包括单机和集群，实例规格请参考[表1-1](#)和[表1-2](#)。

#### 📖 说明

- 为了保证稳定性，服务端限制了单条消息的最大长度为50MB，请勿发送大于此长度的消息。
- 以下表中参考性能，是指以2K大小的消息为例的每秒处理消息条数，测试场景为不开启持久化的非镜像队列，实时生产实时消费，队列无积压。此数据仅供参考，生产使用需要以实际压测性能为准。
- 服务端的性能主要跟以下因素相关：队列数、消息堆积、连接数、channel、消费者数、镜像队列、优先级队列、消息持久化和exchange类型等，在选择实例规格时，请根据业务模型压测结果选择。
- 一条连接最多可以开启2047个channel。

表 1-1 RabbitMQ 单机实例产品规格

实例规格	参考性能 (TPS)	建议队列数	最大连接数
2核   4GB	10000	100	2000

实例规格	参考性能 (TPS)	建议队列数	最大连接数
4核   8GB	20000	200	3000
8核   16GB	35000	400	5000
16核   32GB	45000	800	8000

表 1-2 RabbitMQ 集群实例产品规格

实例规格	代理数	参考性能 (TPS)	建议队列数	最大连接数
4核   8GB	3	45000	600	3000*3
	5	70000	1000	3000*5
	7	80000	1400	3000*7
8核   16GB	3	85000	1200	5000*3
	5	110000	2000	5000*5
	7	120000	2800	5000*7
16核   32GB	3	130000	2400	8000*3
	5	160000	4000	8000*5
	7	180000	5600	8000*7

## RabbitMQ 专享实例的存储空间估算参考

在集群模式中，RabbitMQ需要对消息持久化写入到磁盘中，因为，您在创建 RabbitMQ实例选择存储空间时，建议根据业务消息体积预估以及镜像队列副本数选择选择合适的存储空间。镜像队列副本数最大为集群的代理数。

例如：业务消息体积预估100GB，则磁盘容量最少应为100GB\*镜像队列副本数个数 + 预留磁盘大小100GB。

如果是单机实例，则是计算业务消息体积+预留磁盘大小即可。

当前RabbitMQ实例支持修改集群实例的代理个数，您可以根据业务情况，随时更改集群代理个数。单机实例暂不支持变更规格。

## 1.5 Kafka 与 RabbitMQ 的差异

Kafka采用拉取 ( Pull ) 方式消费消息，吞吐量相对更高，适用于海量数据收集与传递场景，例如日志采集和集中分析。RabbitMQ在吞吐量方面略有逊色，但支持更多的消息队列功能。

以下从性能、数据可靠性、服务可用性、功能等方面给出具体的对比分析，供用户选型参考。

## 性能

消息中间件的性能主要衡量吞吐量，Kafka的吞吐量比RabbitMQ要高出1~2个数量级，RabbitMQ的单机QPS在万级别，Kafka的单机QPS能够达到百万级别。

Kafka如果开启幂等、事务等功能，性能也会有所降低。

## 数据可靠性

Kafka与RabbitMQ都具备多副本机制，数据可靠性较高。

## 服务可用性

Kafka采用集群部署，分区与多副本的设计，使得单代理宕机对服务无影响，且支持消息容量的线性提升。

## 功能

Kafka与RabbitMQ都是比较主流的两款消息中间件，具备消息传递的基本功能，但在一些特殊的功能方面存在差异，以下给出Kafka与RabbitMQ两个开源社区版本的比较。

表 1-3 Kafka 与 RabbitMQ 的功能差异比较

功能项	Kafka ( 1.1.0/2.3.0版本 )	RabbitMQ ( 3.7.17版本 )
优先级队列	不支持	支持。建议优先级大小设置在0-10之间。
延迟队列	不支持	支持
死信队列	不支持	支持
重试队列	不支持	不支持
消费模式	客户端主动拉取	支持客户端主动拉取以及服务端推送两种模式。
广播消费	支持	支持
消息回溯	支持。Kafka支持按照offset和timestamp两种维度进行消息回溯。	不支持。RabbitMQ中消息一旦被确认消费就会被标记删除。
消息堆积	支持。考虑吞吐因素，Kafka的堆积效率比RabbitMQ总体上要高。	支持
持久化	支持	支持
消息追踪	不支持	支持。RabbitMQ中可以采用Firehose或者rabbitmq_tracing插件实现，但开启rabbitmq_tracing插件会影响性能，建议只在定位问题过程中开启。
消息过滤	支持	不支持，但可以自行封装。

功能项	Kafka ( 1.1.0/2.3.0版本 )	RabbitMQ ( 3.7.17版本 )
多租户	不支持	支持
多协议支持	只支持Kafka自定义协议。	RabbitMQ基于AMQP协议实现，同时支持MQTT、STOMP等协议。
跨语言支持	采用Scala和Java编写，支持多种语言的客户端。	采用Erlang编写，支持多种语言的客户端。
流量控制	支持client和user级别，通过主动设置可将流控作用于生产者或消费者。	RabbitMQ的流控基于Credit-Based算法，是内部被动触发的保护机制，作用于生产者层面。
消息顺序性	支持单分区（partition）级别的顺序性。	不支持。需要单线程发送、单线程消费并且不采用延迟队列、优先级队列等一些高级功能整体配合，才能实现消息有序。
安全机制	支持SSL、SASL身份认证和读写权限控制。	与Kafka相似
幂等性	支持单个生产者单分区单会话的幂等性。	不支持
事务性消息	支持	支持

### 📖 说明

以上对比内容仅代表开源版本间的比较。

分布式消息服务提供的Kafka与RabbitMQ，在兼容开源协议的基础上，对版本特性做了部分支持或者增强。

## 1.6 与其他云服务的关系

- 云审计（Cloud Trace Service）  
云审计为您提供云服务资源的操作记录，记录内容包括您从云服务平台管理控制台或者开放API发起的云服务资源操作请求以及每次请求的结果，供您查询、审计和回溯使用。
- 虚拟私有云  
RabbitMQ专享版实例运行于虚拟私有云，需要使用虚拟私有云创建的IP和带宽。通过虚拟私有云安全组的功能可以增强访问RabbitMQ专享版实例的安全性。
- 云监控（Cloud Eye）  
云监控是一个开放性的监控平台，提供资源的实时监控、告警、通知等服务。

### 📖 说明

RabbitMQ专享实例向Cloud Eye上报监控数据的更新周期为1分钟。

## 1.7 约束与限制

分布式消息服务RabbitMQ在某些功能做了约束和限制，如表1-4所示。

表 1-4 RabbitMQ 使用约束和限制

限制项	约束和限制	描述
版本	当前服务端版本为3.7.17	兼容AMQP 0-9-1协议的客户端版本。
连接数	RabbitMQ单机和集群实例，不同实例规格的连接数上限不一致，具体限制，请参考 <a href="#">产品规格</a> 。	-
通道数	<= 2047	单条连接可以建立的通道数。
消息大小	单条消息的最大长度为50MB	服务端限制了单条消息的最大长度为50MB，请勿发送大于此长度的消息，否则生产失败。
内存高水位阈值	<= 40%	内存使用率超过40%会触发内存高水位，生产者流程被阻塞
磁盘高水位阈值	>= 5GB	磁盘剩余空间低于5GB会触发磁盘高水位，生产者流程被阻塞
cluster_partition_handling	pause_minority	当集群发生网络分区时，代理会检查自己是否处于“少数派”（存储分区的代理数小于等于总代理数的一半称为少数派）。少数派中的代理将会自动关闭服务并定期检测网络状态，待分区恢复之后重新启动服务。如果未开启镜像队列，发生分区时少数派上的队列将无法生产消费。 此策略相当于放弃了可用性而选择了数据一致性。
rabbitmq_delayed_message_exchange	插件延迟时间存在1%左右的误差，可能提前或者推迟发送消息给消费者。	实例是否开启消息延迟功能。

## 1.8 RabbitMQ 相关概念

使用RabbitMQ作为消息引擎，RabbitMQ是一个生产者和消费者模型，主要负责接收、存储和转发消息。以下概念基于RabbitMQ进行描述。

## 消息

消息一般分为两部分，消息体和标签，标签主要用来描述这条消息，消息体是消息的内容，是一个json体或者数据等。

生产者发送消息，消费者消费消息，生产者与消费者彼此并无直接关系。

## 生产者 ( Producer )

即向队列发送消息的一方。发布消息的最终目的在于将消息内容传递给其他系统/模块，使对方按照约定处理该消息。

## 消费者 ( Consumer )

接收消息的一方。消费者订阅RabbitMQ的队列，当消费者消费一条消息时，只是消费消息的消息体。在消息路由的过程中，会丢弃标签，存入到队列中的只有消息体。

## 队列 ( Queue )

队列是用于存储消息的，生产者将消息送到队列，消费者从队列中获取和消费消息。多个消费者可以同时订阅同一个队列，队列里的消息分配给不同的消费者。

## 代理 ( Broker )

消息中间件的服务节点。

# 1.9 权限管理

如果您需要对云服务平台上购买的DMS for RabbitMQ资源，给企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，您可以使用统一身份认证服务 ( Identity and Access Management, 简称IAM ) 进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全的控制资源的访问。

通过IAM，您可以在帐号中给员工创建IAM用户，并使用策略来控制他们对资源的访问范围。例如您的员工中有负责软件开发的人员，您希望他们拥有DMS for RabbitMQ的使用权限，但是不希望他们拥有删除RabbitMQ实例等高危操作的权限，那么您可以使用IAM为开发人员创建用户，通过授予仅能使用DMS for RabbitMQ，但是不允许删除RabbitMQ实例的权限策略，控制他们对DMS for RabbitMQ资源的使用范围。

如果帐号已经能满足您的要求，不需要创建独立的IAM用户进行权限管理，您可以跳过本章节，不影响您使用DMS for RabbitMQ服务的其它功能。

## DMS for RabbitMQ 权限

默认情况下，管理员创建的IAM用户没有任何权限，需要将其加入用户组，并给用户组授予策略或角色，才能使得用户组中的用户获得对应的权限，这一过程称为授权。授权后，用户就可以基于被授予的权限对云服务进行操作。

DMS for RabbitMQ部署时通过物理区域划分，为项目级服务。授权时，“作用范围”需要选择“区域级项目”，然后在指定区域对应的项目中设置相关权限，并且该权限仅对此项目生效；如果在“所有项目”中设置权限，则该权限在所有区域项目中都生效。访问DMS for RabbitMQ时，需要先切换至授权区域。

权限根据授权精细程度分为角色和策略。

- **角色**：IAM最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。该机制以服务为粒度，提供有限的服务相关角色用于授权。由于云服务平台各服务之间存在业务依赖关系，因此给用户授予角色时，可能需要一并授予依赖的其他角色，才能正确完成业务。角色并不能满足用户对精细化授权的要求，无法完全达到企业对权限最小化的安全管控要求。
- **策略**：IAM最新提供的一种细粒度授权的能力，可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式，能够满足企业对权限最小化的安全管控要求。例如：针对DMS for RabbitMQ服务，管理员能够控制IAM用户仅能对实例进行指定的管理操作。多数细粒度策略以API接口为粒度进行权限拆分，DMS for RabbitMQ服务支持的API授权项请参见[细粒度策略支持的授权项](#)。

### 📖 说明

DMS for RabbitMQ的权限与策略基于分布式消息服务DMS，因此在IAM服务中为RabbitMQ分配用户与权限时，请选择并使用“DMS”的权限与策略。

如表1-5所示，包括了DMS for RabbitMQ的所有系统权限。

表 1-5 DMS for RabbitMQ 系统权限

系统角色/策略名称	描述	类别	依赖关系
DMS FullAccess	分布式消息服务管理员权限，拥有该权限的用户可以操作所有分布式消息服务的功能。	系统策略	无
DMS UserAccess	分布式消息服务普通用户权限（没有实例创建、修改、删除、扩容）。	系统策略	无
DMS ReadOnlyAccess	分布式消息服务的只读权限，拥有该权限的用户仅能查看分布式消息服务数据。	系统策略	无
DMS Administrator	分布式消息服务的管理员权限。	系统角色	依赖Tenant Guest和VPC Administrator。

表2列出了DMS for RabbitMQ常用操作与系统策略的授权关系，您可以参照该表选择合适的系统策略。

表 1-6 常用操作与系统策略的关系

操作	DMS FullAccess	DMS UserAccess	DMS ReadOnlyAccess
创建实例	√	×	×
修改实例	√	×	×
删除实例	√	×	×

操作	DMS FullAccess	DMS UserAccess	DMS ReadOnlyAccess
变更实例规格	√	×	×
重启实例	√	√	×
查询实例信息	√	√	√



# 2 权限管理

## 2.1 创建用户并授权使用 DMS for RabbitMQ

如果您需要对您所拥有的DMS for RabbitMQ服务进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称IAM），通过IAM，您可以：

- 根据企业的业务组织，在您的帐号中，给企业中不同职能部门的员工创建IAM用户，让员工拥有唯一安全凭证，并使用DMS for RabbitMQ资源。
- 根据企业用户的职能，设置不同的访问权限，以达到用户之间的权限隔离。
- 将DMS for RabbitMQ资源委托给更专业、高效的其他帐号或者云服务，这些帐号或者云服务可以根据权限进行代运维。

如果帐号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用DMS for RabbitMQ服务的其它功能。

本章节以创建用户并授予“DMS ReadOnlyAccess”权限为例，为您介绍对用户授权的方法，操作流程如[图2-1](#)所示。

### 说明

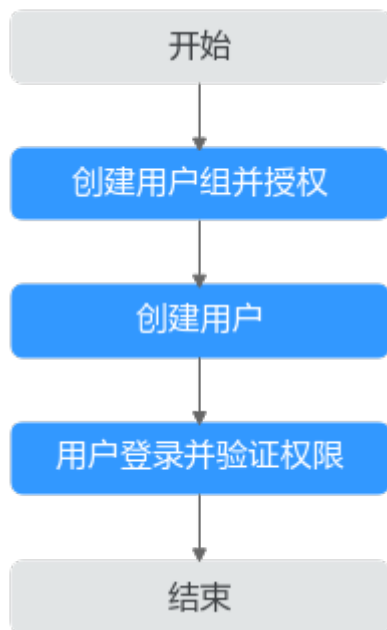
DMS for RabbitMQ服务的权限与策略基于分布式消息服务DMS，因此在IAM服务中为RabbitMQ分配用户与权限时，请选择并使用“DMS”的权限与策略。

### 前提条件

给用户组授权之前，请您了解用户组可以添加的DMS for RabbitMQ系统策略，并结合实际需求进行选择，DMS for RabbitMQ支持的系统策略及策略间的对比，请参见：产品简介 > 权限管理。

## 示例流程

图 2-1 给用户授权 DMS for RabbitMQ 权限流程



1. 创建用户组并授权  
在IAM控制台创建用户组，并授予DMS for RabbitMQ的只读权限“DMS ReadOnlyAccess”。
2. 创建用户并加入用户组  
在IAM控制台创建用户，并将其加入1中创建的用户组。
3. 用户登录并验证权限  
新创建的用户登录控制台，验证DMS for RabbitMQ的只读权限。

## 2.2 DMS for RabbitMQ 自定义策略

如果系统预置的DMS for RabbitMQ权限，不满足您的授权要求，可以创建自定义策略。自定义策略中可以添加的授权项（Action）请参考[细粒度策略支持的授权项](#)。

目前云服务平台支持以下两种方式创建自定义策略：

- 可视化视图创建自定义策略：无需了解策略语法，按可视化视图导航栏选择云服务、操作、资源、条件等策略内容，可自动生成策略。
- JSON视图创建自定义策略：可以在选择策略模板后，根据具体需求编辑策略内容；也可以直接在编辑框内编写JSON格式的策略内容。

具体创建步骤请参见：《统一身份认证服务 用户指南》的“创建自定义策略”章节。本章为您介绍常用的DMS for RabbitMQ自定义策略样例。

### 📖 说明

DMS for RabbitMQ服务的权限与策略基于分布式消息服务DMS，因此在IAM服务中为RabbitMQ分配用户与权限时，请选择并使用“DMS”的权限与策略。

## DMS for RabbitMQ 自定义策略样例

- 示例1：授权用户删除实例和重启实例

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "dms:instance:delete",
        "dms:instance:modifyStatus"
      ]
    }
  ]
}
```

- 示例2：拒绝用户删除实例

拒绝策略需要同时配合其他策略使用，否则没有实际作用。用户被授予的策略中，一个授权项的作用如果同时存在Allow和Deny，则遵循Deny优先。

如果您给用户授予DMS FullAccess的系统策略，但不希望用户拥有DMS FullAccess中定义的删除实例权限，您可以创建一条拒绝删除实例的自定义策略，然后同时将DMS FullAccess和拒绝策略授予用户，根据Deny优先原则，则用户可以对DMS FullAccess执行除了删除实例外的所有操作。拒绝策略示例如下：

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "dms:instance:delete"
      ]
    }
  ]
}
```


# 3 环境准备

创建RabbitMQ实例前，您需要创建虚拟私有云（Virtual Private Cloud，以下简称VPC），并且已配置好安全组与子网。VPC为RabbitMQ实例提供一个隔离的、用户自主配置和管理的虚拟网络环境，提升云服务资源的安全性，简化用户的网络部署。

如果用户已有VPC，可重复使用，不需要多次创建。


## 创建 VPC

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“网络 > 虚拟私有云”。

**步骤4** 单击“创建虚拟私有云”，进入“创建虚拟私有云”界面。

**步骤5** 根据界面提示创建虚拟私有云。如无特殊需求，界面参数均可保持默认。关于创建VPC的详细信息可以参考《虚拟私有云用户指南》。

创建虚拟私有云时，会同时创建子网，若需要额外创建子网，请参考**步骤6**。如果不需要额外创建子网，请执行**步骤7**。

**步骤6** 在左侧导航栏，单击“子网”，进入“子网”界面。单击“创建子网”。根据界面提示创建子网。如无特殊需求，界面参数均可保持默认。

关于创建子网的详细信息可以参考《虚拟私有云用户指南》。

**步骤7** 在左侧导航栏，选择“访问控制 > 安全组”，进入“安全组”界面。根据界面提示创建安全组。如无特殊需求，界面参数均可保持默认。

关于创建安全组的详细信息可以参考《虚拟私有云用户指南》。

----结束

# 4 购买实例

## 操作场景

目前，DMS提供RabbitMQ类型的队列服务，RabbitMQ实例采用物理隔离的方式部署，租户独占RabbitMQ实例。支持用户自定义规格和自定义特性，您可以根据业务需要定制相应计算能力和存储空间的RabbitMQ实例。

RabbitMQ是一款基于AMQP协议的开源服务，用于在分布式系统中存储转发消息，服务器端用Erlang语言（支持高并发、分布式以及健壮的容错能力等特点）编写，支持多种语言的客户端，如：Python、Ruby、.NET、Java、JMS、C、PHP、ActionScript、XMPP、STOMP、AJAX等。


AMQP，即Advanced Message Queuing Protocol，高级消息队列协议，是应用层的一个开放标准协议，为面向消息的中间件设计，主要特征为面向消息、队列、路由、可靠性、安全。

## 前提条件

RabbitMQ实例运行于虚拟私有云，购买实例前，需保证有可用的虚拟私有云，并且已配置好安全组与子网。


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击页面右上方的“购买RabbitMQ实例”。

**步骤5** 选择“计费模式”、“区域”、“项目”和“可用区”。

**步骤6** 设置“实例名称”和“企业项目”。

### 步骤7 设置实例信息。

1. 版本：RabbitMQ的版本号，当前仅支持RabbitMQ 3.7.17。
2. 实例类型：支持“单机”和“集群”。
  - 单机：表示部署一个RabbitMQ代理。
  - 集群：表示部署多个RabbitMQ代理，实现高可靠的消息存储。
3. CPU架构：当前仅支持“x86计算”，保持默认值即可。
4. 规格：根据实际情况选择规格。

#### 说明

为了保证服务的稳定可靠，RabbitMQ采用了默认的40%高水位配置。当内存占用率到达40%高水位后，会触发流控，生产者发送消息会被阻塞。为了避免高水位的产生，请及时消费积压在队列中的消息。

5. 存储空间：选择RabbitMQ实例的磁盘类型和储存空间总量。
  - 单机实例的取值范围200GB ~ 90000GB。
  - 集群实例的取值范围为：100GB\*代理数 ~ 90000GB、200GB\*代理数 ~ 90000GB、300GB\*代理数 ~ 90000GB。
6. 虚拟私有云：选择已经创建好的虚拟私有云和子网。

虚拟私有云可以为您的RabbitMQ实例构建隔离的、能自主配置和管理的虚拟网络环境。
7. 安全组：选择已经创建好的安全组。

安全组是一组对弹性云服务器的访问规则的集合，为同一个VPC内具有相同安全保护需求并相互信任的弹性云服务器提供访问策略。

您可以单击右侧的“管理安全组”，跳转到网络控制台的安全组页面，查看或创建安全组。

### 步骤8 设置连接RabbitMQ实例的用户名和密码。

### 步骤9 单击“更多配置”，设置更多相关信息。

1. 设置“公网访问”。

您可以选择是否打开公网访问开关。

如果选择了开启，表示访问RabbitMQ实例可以通过弹性IP访问。这时页面会显示“弹性IP地址”，在“弹性IP地址”区域，您可下拉选择已有的弹性IP。另外，您可单击右侧的“创建弹性IP”，跳转到网络控制台的弹性公网IP页面，创建弹性公网IP。

#### 说明

- 公网访问与VPC内访问相比，可能存在网络丢包和抖动等情况，且访问时延有所增加，因此建议仅在业务开发测试阶段开启公网访问进行调试。
  - 如果用户在虚拟私有云的服务页面手动解绑定或删除EIP，DMS服务会自动关闭相应RabbitMQ实例的公网访问功能。
2. 设置“SSL”。

客户端连接实例时SSL认证的开关。开启SSL，则数据加密传输，安全性更高。

**SSL开关在实例创建完成后不支持修改，请明确是否需要开启。**
  3. 设置“标签”。

标签用于标识云资源，当您拥有相同类型的许多云资源时，可以使用标签按各种维度（例如用途、所有者或环境）对云资源进行分类。

- 如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键值对。另外，您可以单击右侧的“查看预定义标签”，系统会跳转到标签管理服务页面，查看已经预定义的标签，或者创建新的标签。
- 您也可以直接在“标签键”和“标签值”中设置标签。

当前每个RabbitMQ实例最多支持设置20个不同标签，标签的命名规格，请参考[管理实例标签](#)章节。

#### 4. 设置实例的描述信息。

**步骤10** 填写完上述信息后，单击“立即购买”，进入“规格确认”页面。

**步骤11** 确认实例信息无误后，单击“提交”。

**步骤12** 在实例列表页面查看实例是否创建成功。

创建实例大约需要3到15分钟，此时实例的“状态”为“创建中”。

- 当实例的“状态”变为“运行中”时，说明实例创建成功。
- 如果创建实例失败，在信息栏的“创建失败任务”中查看创建失败的实例。请参考[删除实例](#)，删除创建失败的RabbitMQ实例，然后重新购买。如果重新购买仍然失败，请联系客服。

----结束

# 5 连接实例

## 5.1 连接未开启 SSL 方式的 RabbitMQ 实例

RabbitMQ实例兼容开源协议，请参考RabbitMQ官网提供的不同语言的连接和使用向导：<https://www.rabbitmq.com/getstarted.html>。

本节以分布式消息服务RabbitMQ版提供的demo为例，介绍VPC内访问与使用RabbitMQ的方法，假设RabbitMQ客户端部署在弹性云服务器上。

如果RabbitMQ实例开启了SSL认证开关，连接方式请参考[连接已开启SSL方式的RabbitMQ实例](#)。

### 前提条件

- 参考[购买实例](#)章节创建RabbitMQ实例，并记录创建时输入的用户名和密码。
- 创建完成后，单击实例名称，查看并记录实例详情中的“连接地址”。
- 已创建弹性云服务器，并且弹性云服务器的VPC、子网、安全组与RabbitMQ实例的VPC、子网、安全组保持一致。

### 命令行模式连接实例

**步骤1** 登录弹性云服务器。

**步骤2** 安装Java JDK或JRE，并配置JAVA\_HOME与PATH环境变量。在用户家目录下修改.bash\_profile，添加如下行，路径以实际为准。

```
export JAVA_HOME=/opt/java/jdk1.8.0_151
export PATH=$JAVA_HOME/bin:$PATH
```

执行source .bash\_profile命令使修改生效。

#### 说明

ECS虚拟机默认自带的JDK可能不符合要求，例如OpenJDK，需要配置为Oracle的JDK，可至Oracle官方下载页面[下载Java Development Kit 1.8.111及以上版本](#)。

**步骤3** 下载RabbitMQ-Tutorial.zip示例工程代码。

```
$ wget https://dms-demo.obs.cn-north-1.myhuaweicloud.com/RabbitMQ-Tutorial.zip
```



**步骤4** 解压RabbitMQ-Tutorial.zip压缩包。

```
$ unzip RabbitMQ-Tutorial.zip
```

**步骤5** 进入RabbitMQ-Tutorial目录，该目录下包含预编译好的jar文件。

```
$ cd RabbitMQ-Tutorial
```

**步骤6** 运行生产消息示例。

```
$ java -cp ./rabbitmq-tutorial.jar Send host port user password
```

其中，host表示RabbitMQ实例的连接地址，port为RabbitMQ实例的监听端口（默认为5672），user表示RabbitMQ用户名，password表示用户名对应的密码。

图 5-1 运行生产消息示例

```
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java ./rabbitmq-tutorial.jar Send 192.168.0.37 5672 admin admin
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.0.37 5672 admin admin
[x] Sent 'Hello World!'
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.0.37 5672 admin admin
[x] Sent 'Hello World!'
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.0.37 5672 admin admin
[x] Sent 'Hello World!'
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.0.37 5672 admin admin
[x] Sent 'Hello World!'
```

使用Ctrl+C命令退出。

**步骤7** 运行消费消息示例。

```
$ java -cp ./rabbitmq-tutorial.jar Recv host port user password
```

其中，host表示RabbitMQ实例的连接地址，port为RabbitMQ实例的监听端口（默认为5672），user表示RabbitMQ用户名，password表示用户名对应的密码。

图 5-2 运行消费消息示例

```
[root@rabbitmq-0004 RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Recv 192.168.0.37 5672 admin admin
[*] Waiting for messages. To exit press CTRL+C
[x] Received 'Hello World!'
[x] Received 'Hello World!'
[x] Received 'Hello World!'
[x] Received 'Hello World!'
```

如需停止消费使用Ctrl+C命令退出。

----结束

## 示例代码（Java）

### 连接实例并生产消息

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(host);
factory.setPort(port);

factory.setUsername(user);
factory.setPassword(password);
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();

channel.queueDeclare(QueueName, false, false, false, null);

String message = "Hello World!";
channel.basicPublish("", QueueName, null, message.getBytes("UTF-8"));
System.out.println(" [x] Sent '" + message + "'");

channel.close();
connection.close();
```

### 连接实例并消费消息

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(host);
factory.setPort(port);
factory.setUsername(user);
factory.setPassword(password);
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();

channel.queueDeclare(QueueName, false, false, false, null);
System.out.println(" [*] Waiting for messages. To exit press CTRL+C");

Consumer consumer = new DefaultConsumer(channel)
{
    @Override
    public void handleDelivery(String consumerTag, Envelope envelope, AMQP.BasicProperties properties,
        byte[] body)
        throws IOException
    {
        String message = new String(body, "UTF-8");
        System.out.println(" [x] Received '" + message + "'");
    }
};
channel.basicConsume(QueueName, true, consumer);
```

## 5.2 连接已开启 SSL 方式的 RabbitMQ 实例

创建实例时开启SSL访问，则数据加密传输，安全性更高。

本节介绍VPC内访问开启SSL的RabbitMQ实例的方法。

### 前提条件

- 参考[购买实例](#)章节创建RabbitMQ实例，并记录创建时输入的用户名和密码。
- 创建完成后，单击实例名称，查看并记录实例详情中的“连接地址”。
- 已创建弹性云服务器，并且弹性云服务器的VPC、子网、安全组与RabbitMQ实例的VPC、子网、安全组保持一致。

### 命令行模式连接实例

**步骤1** 登录弹性云服务器，如开启公网访问，则直接登录执行主机。

**步骤2** 安装Java JDK或JRE，并配置JAVA\_HOME与PATH环境变量，使用执行用户在用户家目录下修改.bash\_profile，添加如下行，路径以实际为准。

```
export JAVA_HOME=/opt/java/jdk1.8.0_151
export PATH=$JAVA_HOME/bin:$PATH
```

执行source .bash\_profile命令使修改生效。

#### 📖 说明

ECS虚拟机默认自带的JDK可能不符合要求，例如OpenJDK，需要配置为Oracle的JDK，可至Oracle官方下载页面[下载Java Development Kit 1.8.111及以上版本](#)。

**步骤3** 下载RabbitMQ-Tutorial-SSL.zip示例工程代码。

```
$ wget https://dms-demo.obs.cn-north-1.myhuaweicloud.com/RabbitMQ-Tutorial-SSL.zip
```

**步骤4** 解压RabbitMQ-Tutorial-SSL.zip压缩包。

```
$ unzip RabbitMQ-Tutorial-SSL.zip
```

**步骤5** 进入RabbitMQ-Tutorial-SSL目录，该目录下包含预编译好的jar文件。

```
$ cd RabbitMQ-Tutorial-SSL
```

**步骤6** 运行生产消息示例。

```
$ java -cp ../rabbitmq-tutorial-sll.jar Send host port user password
```

其中，host表示RabbitMQ实例的连接地址，port为RabbitMQ实例的监听端口（默认为5671），user表示RabbitMQ用户名，password表示用户名对应的密码。

图 5-3 生产消息示例

```
root@ecs-3b6f RabbitMQ-Tutorial-SSL# java -cp ../rabbitmq-tutorial-sll.jar Send 192.168.1.35 5671 root admin123
LF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
LF4J: Defaulting to no-operation (NOP) logger implementation
LF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
[x] Sent 'Hello World!'
root@ecs-3b6f RabbitMQ-Tutorial-SSL# java -cp ../rabbitmq-tutorial-sll.jar Send 192.168.1.35 5671 root admin123
LF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
LF4J: Defaulting to no-operation (NOP) logger implementation
LF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
[x] Sent 'Hello World!'
```

使用Ctrl+C命令退出。

**步骤7** 运行消费消息示例。

```
$ java -cp ../rabbitmq-tutorial-sll.jar Recv host port user password
```

其中，host表示RabbitMQ实例的连接地址，port为RabbitMQ实例的监听端口（默认为5671），user表示RabbitMQ用户名，password表示用户名对应的密码。

图 5-4 消费消息示例

```
root@ecs-3b6f RabbitMQ-Tutorial-SSL# java -cp ../rabbitmq-tutorial-sll.jar Recv 192.168.1.35 5671 root admin123
LF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
LF4J: Defaulting to no-operation (NOP) logger implementation
LF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
[*] Waiting for messages. To exit press CTRL+C
[x] Received 'Hello World!'
[x] Received 'Hello World!'
C[root@ecs-3b6f RabbitMQ-Tutorial-SSL#
```

如需停止消费使用Ctrl+C命令退出。

----结束

## 示例代码（Java）

连接实例并生产消息

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(host);
factory.setPort(port);

factory.setUsername(user);
factory.setPassword(password);
factory.useSslProtocol();
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();

channel.queueDeclare(QUEUE_NAME, false, false, false, null);

String message = "Hello World!";
channel.basicPublish("", QUEUE_NAME, null, message.getBytes("UTF-8"));
System.out.println(" [x] Sent '" + message + "'");

channel.close();
connection.close();
```

连接实例并消费消息

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(host);
factory.setPort(port);
factory.setUsername(user);
factory.setPassword(password);
factory.useSslProtocol();
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();

channel.queueDeclare(QUEUE_NAME, false, false, false, null);
System.out.println(" [*] Waiting for messages. To exit press CTRL+C");


Consumer consumer = new DefaultConsumer(channel)
{
    @Override
    public void handleDelivery(String consumerTag, Envelope envelope, AMQP.BasicProperties properties,
        byte[] body)
        throws IOException
    {
        String message = new String(body, "UTF-8");
        System.out.println(" [x] Received '" + message + "'");
    }
};
channel.basicConsume(QUEUE_NAME, true, consumer);
```

## 5.3 连接 RabbitMQ 管理地址

需要通过浏览器输入RabbitMQ管理地址，可以访问开源RabbitMQ的集群管理工具。


### 操作步骤

**步骤1** 获取实例管理地址。

1. 登录管理控制台。
2. 在管理控制台左上角单击 ，选择区域。

#### 说明

此处请选择与您的应用服务相同的区域。

3. 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。
4. 单击实例名称，进入实例详情页面，获取Web界面UI地址和用户名。

#### 说明

用户名和密码为创建RabbitMQ实例时自定义的内容。

**步骤2** 确认实例安全组规则是否配置正确。

1. 在实例详情页面的“基本信息 > 网络”，单击安全组名称，跳转到安全组页面。
2. 选择“入方向规则”，查看安全组入方向规则。
  - 实例未开启SSL开关
    - 如果是VPC内访问，实例安全组入方向规则，需要允许端口5672的访问。
    - 如果是公网访问，需要允许端口15672的访问。
  - 实例已开启SSL开关

- 如果是VPC内访问，实例安全组入方向规则，需要允许端口5671的访问。
- 如果是公网访问，需要运行端口15671的访问。

**步骤3** 在浏览器中打开Web界面UI地址，进入Web登录页面。

#### 📖 说明

- 如果RabbitMQ实例开启了公网访问，可直接在公网环境下的浏览器访问Web页面。
- 如果RabbitMQ实例未开启公网访问，您需要购买一台与RabbitMQ实例网络相通的Windows弹性云服务器，然后登录弹性云服务器访问Web页面。

图 5-5 登录实例 Web 页面



**步骤4** 单击“Login”，登录完成。

----结束

## 5.4 开启心跳

客户端连接RabbitMQ集群实例时，如果存在消息收发时间间隔大于90秒的场景，请在客户端开启心跳并设置小于90秒的心跳超时时间，防止断连。

### 什么是心跳

RabbitMQ实例提供了心跳功能，以确保应用程序层及时发现中断的连接和完全无响应的对端。心跳还可以防止某些网络设备在一段时间内由于没有活动而中断TCP连接。**开启心跳的方法为在连接上指定心跳超时时间。**

心跳超时时间定义了对等TCP连接在多长时间后被服务端和客户端视为关闭。服务端和客户端会对配置的心跳超时时间进行协商，客户端必须配置该值来发送心跳。RabbitMQ官方团队维护的3个客户端（Java、.NET、Erlang语言）的心跳超时时间协商逻辑如下：

- 服务端和客户端设置的心跳超时时间都不为0时，两者间较小的值生效。
- 服务端和客户端任意一端设置的心跳超时时间为0，另一端不为0时，非0的值生效。
- 服务端和客户端的心跳超时时间都设置为0时，表示禁用心跳。

配置心跳超时时间后，RabbitMQ服务端和客户端都会向对方发送AMQP心跳帧作为心跳，发送的时间间隔为心跳超时时间的一半。客户端在两次错过心跳后，会被认为是

不可达的，TCP连接将被关闭。当客户端检测到服务端由于心跳而无法访问时，需要重新连接。

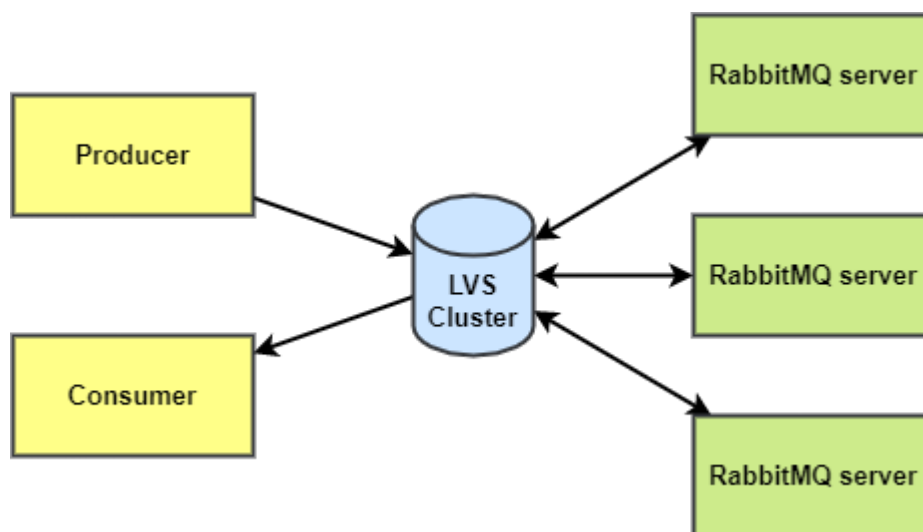
### 须知

一些客户端（如C语言客户端）没有发送心跳的逻辑，即使配置了心跳超时时间，开启了心跳，仍然无法发送心跳。此时需要额外启动一个线程，编写发送心跳的逻辑。

## LVS 的心跳超时时间

RabbitMQ集群实例使用LVS进行负载均衡，如图5-6所示，单节点实例不涉及LVS。

图 5-6 集群实例的负载均衡



LVS对客户端连接设置了心跳超时时间，默认为90秒。如果客户端在90秒内没有向LVS发送心跳（AMQP心跳帧或消息收发），LVS会主动断开与客户端的连接，此时客户端需要重新连接。

如果存在消息收发时间间隔大于90秒的场景，请在客户端开启心跳并设置小于90秒的心跳超时时间。

## 客户端如何配置心跳超时时间

- 在Java客户端配置心跳超时时间。  
在创建连接前使用**ConnectionFactory#setRequestedHeartbeat**进行设置，示例如下：

```
ConnectionFactory cf = new ConnectionFactory();
// 将心跳超时时间设置为60秒
cf.setRequestedHeartbeat(60);
```

- 在.NET客户端配置心跳超时时间，示例如下。

```
var cf = new ConnectionFactory();
// 将心跳超时设置为60秒
cf.RequestedHeartbeat = TimeSpan.FromSeconds(60);
```

- 在Python pika客户端配置心跳超时时间，示例如下。

```
# 设置心跳时间为60秒
params = pika.ConnectionParameters(host='host', heartbeat=60,
```

```
credentials=pika.PlainCredentials('username', 'passwd')
connection = pika.BlockingConnection(params)

while True:
    channel.basic_publish(exchange="", routing_key='hello', body='Hello World!')
    print(" [x] Sent 'Hello World!'")
    # 生产者需要使用connection.sleep()才能触发心跳, 使用时间.sleep()不会触发心跳
    connection.sleep(200)
```

## 5.5 查看客户端连接地址

分布式消息服务RabbitMQ版支持通过Web界面查看客户端连接地址。

### 说明

客户端处于连接RabbitMQ实例时，才可以查看客户端连接地址。

### 操作步骤

- 步骤1 [登录RabbitMQ Web界面](#)。
- 步骤2 在导航栏单击“Connections”，进入“Connections”页面。
- 步骤3 查看客户端连接地址，如[图5-7](#)所示。

图 5-7 客户端连接地址

The screenshot shows the 'Connections' page in the RabbitMQ management interface. It displays a table with 4 connections. The first three connections have IP addresses highlighted in red: 10.234.177.66:50996, 10.234.177.66:53332, and 10.234.177.66:56272. The fourth connection has IP address 172.31.1.152:5004. The table columns include Name, Node, User name, State, SSL / TLS, Protocol, Channels, From client, and To client.

Overview		Details			Network			
Name	Node	User name	State	SSL / TLS	Protocol	Channels	From client	To client
10.234.177.66:50996	rabbit@dms-vm-4cd31738-rabbitmq-1	root	running	•	AMQP 0-9-1	1	0iB/s	0iB/s
10.234.177.66:53332	rabbit@dms-vm-4cd31738-rabbitmq-1	root	running	•	AMQP 0-9-1	1	0iB/s	0iB/s
10.234.177.66:56272	rabbit@dms-vm-4cd31738-rabbitmq-2	root	running	•	AMQP 0-9-1	1	0iB/s	0iB/s
172.31.1.152:5004	rabbit@dms-vm-4cd31738-rabbitmq-0	root	running	•	AMQP 0-9-1	1	0iB/s	0iB/s

同一个客户端可以作为生产者生产消息，也可以作为消费者消费消息，连接IP地址是相同的，如[图5-7](#)所示，此时我们无法区分哪个是生产者IP地址，哪个是消费者IP地址。如果想要直观体现生产者/消费者IP地址，您可以在客户端中设置“clientProperties”参数，通过此参数来标明生产者/消费者IP地址，示例如下。

```
//配置客户端连接参数
HashMap<String, Object> clientProperties = new HashMap<>();
clientProperties.put("connection_name", "producer");
connectionFactory.setClientProperties(clientProperties);

//创建连接
Connection connection = connectionFactory.newConnection();
```

设置“clientProperties”参数后，连接地址显示如[图5-8](#)所示。

图 5-8 客户端连接地址（分区生产者/消费者 IP 地址）

Connections

▼ All connections (2)

Pagination

Page 1 of 1 - Filter:   Regex ?

Overview			Details			Network				+/-
Name	User name	State	SSL / TLS	Protocol	Channels	From client	To client	Heartbeat	Connected at	
10.234.177.66:65260 consumer	admin	running	o	AMQP 0-9-1	1	0iB/s	0iB/s	60s	10:53:21 2022-07-13	
10.234.177.66:58373 producer	admin	running	o	AMQP 0-9-1	1	0iB/s	0iB/s	60s	10:44:16 2022-07-13	

[HTTP API](#) [Server Docs](#) [Tutorials](#) [Community Support](#) [Community Slack](#) [Commercial Support](#) [Plugins](#) [GitHub](#)

----结束



# 6 实例日常操作

## 6.1 查看实例

### 操作场景


本节介绍如何在控制台查看RabbitMQ实例的详细信息。例如，连接RabbitMQ时，需要获取连接IP和端口。

### 前提条件

已创建RabbitMQ实例。


### 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

#### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** RabbitMQ实例支持通过筛选来查询对应的RabbitMQ实例。当前支持的筛选条件为“项目”、“状态”、“名称”、“连接地址”和“ID”。RabbitMQ实例状态请参见 [表6-1](#)。

表 6-1 RabbitMQ 实例状态说明

状态	说明
创建中	创建RabbitMQ实例后，在RabbitMQ实例状态进入运行中之前的状态。

状态	说明
运行中	RabbitMQ实例正常运行状态。 在这个状态的实例可以运行您的业务。
故障	RabbitMQ实例处于故障的状态。
启动中	RabbitMQ实例从已冻结到运行中的中间状态。
重启中	RabbitMQ实例正在进行重启操作。
变更中	RabbitMQ专享版实例正在进行规格变更操作。
变更失败	RabbitMQ专享版实例处于规格变更操作失败的状态。
已冻结	RabbitMQ实例处于已冻结状态。
冻结中	RabbitMQ实例从运行中到已冻结的中间状态。
升级中	RabbitMQ实例正在进行升级操作。
回滚中	RabbitMQ实例正在进行回滚操作。

**步骤5** 单击RabbitMQ实例的名称，进入该RabbitMQ实例的基本信息页面，查看RabbitMQ实例的详细信息。

表 6-2 实例参数

信息类型	参数	说明
实例信息	实例名称	RabbitMQ实例的名称。单击“名称”后的  可以修改实例名称。
	状态	RabbitMQ实例的状态。
	实例类型	RabbitMQ实例的类型，当前支持单机和集群实例。
	规格	RabbitMQ实例规格。
	Web界面UI地址	未开启公网访问时，连接RabbitMQ集群管理工具的地址。
	维护时间窗	运维操作时间。单击参数后的  可以修改时间窗。
	企业项目	实例的企业项目。单击参数后的  可以修改实例的企业项目。
	实例ID	RabbitMQ实例ID。
	版本	RabbitMQ的版本。当前支持3.7.17。
	创建时间	RabbitMQ实例的创建时间。

信息类型	参数	说明
	用户名	登录RabbitMQ集群管理工具的用户名。当您忘记密码时，可以单击右侧的“重置密码”修改密码。
	描述	RabbitMQ实例的描述信息。单击“描述”后的  可以修改描述信息。
存储信息	已用/可用存储空间 (GB)	RabbitMQ实例已经使用的存储空间和您可以使用的最大存储空间。
	磁盘类型	实例的磁盘类型。当前支持高I/O和超高I/O。
公网访问	公网访问	是否开启公网访问开关。单击  , 修改公网特性。
	公网访问地址	RabbitMQ实例公网连接地址。
	公网访问web界面UI地址	公网访问RabbitMQ集群管理工具的地址。
网络	可用区	RabbitMQ所属的可用区。
	安全组	RabbitMQ实例所关联的安全组。
	虚拟私有云	实例所在的私有网络。
	子网	实例所属子网。

---结束

## 6.2 重启实例

### 操作场景

分布式消息服务RabbitMQ版管理控制台支持重启运行中的RabbitMQ实例，且可实现批量重启RabbitMQ实例。

#### 须知


在RabbitMQ实例重启过程中，客户端的生产与消费消息等请求会被拒绝。

### 前提条件

只有当RabbitMQ实例处于“运行中”或“故障”状态，才能执行重启操作。


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 通过以下任何一种方法，重启RabbitMQ实例。

- 勾选RabbitMQ实例名称左侧的方框，可选一个或多个，单击信息栏左上侧的“重启”。
- 在待重启RabbitMQ实例所在行，单击“重启”。

**步骤5** 单击“是”，完成重启RabbitMQ实例。

重启RabbitMQ实例大约需要3到15分钟。RabbitMQ实例重启成功后，RabbitMQ实例状态切换为“运行中”。

### 说明

重启RabbitMQ只会重启实例进程，不会重启实例所在虚拟机。

如果只需要重启单个RabbitMQ实例，也可以在“RabbitMQ专享版”界面，单击指定RabbitMQ实例右侧“操作”栏下的“重启”。

----结束

## 6.3 删除实例

### 操作场景

分布式消息服务RabbitMQ版管理控制台支持删除RabbitMQ实例，且可实现批量删除RabbitMQ实例、一键式删除创建失败的RabbitMQ实例。

#### 须知


RabbitMQ实例删除后，实例中原有的数据将被删除，且没有备份，请谨慎操作。

### 前提条件

RabbitMQ实例状态为运行中、故障时，才能执行删除操作。


### 删除 RabbitMQ 实例

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 📖 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 通过以下任意一种方法，删除RabbitMQ实例。

- 勾选RabbitMQ实例名称左侧的方框，可选一个或多个，单击信息栏左上侧的“删除”。
- 在待删除RabbitMQ实例所在行，单击“更多 > 删除”。

### 📖 说明

RabbitMQ实例状态为创建中、启动中、变更中、变更失败、重启中时不允许执行删除操作。


**步骤5** 单击“是”，完成删除RabbitMQ实例。

删除RabbitMQ实例大约需要1到60秒。

----结束


## 删除创建失败的 RabbitMQ 实例

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 📖 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

若当前存在创建失败的RabbitMQ实例，界面信息栏会显示“创建失败任务”及失败数量信息。

**步骤4** 单击“创建失败任务”后的图标或者数量，弹出“创建失败任务”对话框。

**步骤5** 在“创建失败任务”界面删除创建失败的RabbitMQ实例。

- 单击“清理失败任务”，一键式删除所有创建失败的RabbitMQ实例。
- 单击需要删除的RabbitMQ实例右侧的“删除任务”，依次删除创建失败的RabbitMQ实例。


----结束

## 6.4 修改实例信息

创建RabbitMQ实例成功后，您可以根据自己的业务情况对RabbitMQ实例的部分参数进行调整。


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击RabbitMQ实例的名称，进入实例详情页面。

**步骤5** 以下参数支持修改。

- 实例名称
- 维护时间窗
- 描述
- 企业项目
- 安全组
- 公网访问（公网访问的修改方法，请参考[设置实例的公网访问](#)。）

参数修改完成后，通过以下方式查看修改结果。

- 修改“公网访问”，系统跳转到“后台任务管理”页签，并显示当前任务的操作进度和结果。
- 修改“实例名称”、“维护时间窗”、“描述”、“企业项目”和“安全组”后，右上角直接提示修改结果。

----结束

## 6.5 重置实例密码

### 操作场景


用户忘记了创建实例时设置的密码，通过重置密码功能，重新设置一个新的密码，可使用新密码连接RabbitMQ实例。

### 说明

只有处于“运行中”状态的RabbitMQ实例支持重置密码。


### 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 在待重置密码的实例所在行，单击“更多 > 重置密码”。

**步骤5** 输入“新密码”和“确认密码”，单击“确定”完成密码重置。

- 重置密码成功，界面提示重置实例的密码成功。
- 重置密码失败，界面提示重置实例的密码失败，请重新尝试重置密码操作。如果多次重置失败，请联系客服处理。

#### 说明

只有所有代理都重置密码成功，才会提示重置密码成功，否则会提示重置失败。

---结束

## 6.6 变更实例规格

### 操作场景

RabbitMQ集群实例创建成功后，支持在变更规格中扩容实例集群代理个数，不支持规格大小变更。单机RabbitMQ实例不支持变更规格。

#### 须知


- 为了实例运行正常，变更规格过程中，请勿对实例进行其他操作。
- 实例规格变更过程中会重启LVS节点，客户端连接会断开，因此业务代码需要对此场景做处理，否则会造成业务中断。

### 前提条件

已创建RabbitMQ实例，且实例状态为“运行中”。


### 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

#### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 在实例所在行，单击“更多 > 变更规格”，进入“分布式消息服务RabbitMQ版变更规格”页面。

**步骤5** 在“节点数”中，选择扩容后的集群节点数，单击“提交”。

在“RabbitMQ专享版”页面的“规格”中查看扩容后的节点个数。

----结束

## 6.7 设置实例的公网访问

### 操作场景

当用户需要通过公网访问RabbitMQ实例时，需要开启该实例的公网访问功能。当业务不再使用公网访问功能时，也可以关闭实例的公网访问功能。

#### 须知


- 仅状态为运行中的实例，可以开启公网访问功能。
- 公网访问与VPC内访问相比，可能存在网络丢包和抖动等情况，且访问时延有所增加，因此建议仅在业务开发测试阶段开启公网访问RabbitMQ实例。
- 开启公网访问后，有如下注意事项：
  - 如果实例未开启SSL  
修改实例的安全组策略，增加入方向规则，允许端口5672和15672的访问。  
访问RabbitMQ管理面：输入地址`http://{RabbitMQ实例公网IP地址}:15672`，然后输入自己配置的用户名和密码。  
Client方式：请使用5672端口。
  - 如果实例开启SSL  
修改实例的安全组策略，增加入方向规则，允许端口5671和15671的访问。  
访问RabbitMQ管理面：输入地址`https://{RabbitMQ实例公网IP地址}:15671`，然后输入自己配置的用户名和密码。  
Client方式：请使用5671端口。

### 前提条件

已创建RabbitMQ实例。


### 开启公网访问

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

#### 说明


此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击待开启公网访问的实例名称，进入实例详情页面。



**步骤5** 单击“公网访问”右侧的 ，打开公网访问开关。

**步骤6** 从“弹性IP地址”下拉列表中选择一个弹性IP，然后单击 ，开启公网访问功能。


如果“弹性IP地址”下拉列表没有值，可单击“创建弹性IP”，跳转到弹性公网IP页面，您可以申请一个新的弹性IP。

开启公网访问功能大约需要10~30秒，请耐心等待。开启公网访问后，页面会自动跳转到“后台任务管理”页签，当任务状态为“成功”时，表示开启公网访问成功。

----结束


## 关闭公网访问

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。


### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击待关闭公网访问的实例名称，进入实例详情页面。

**步骤5** 单击“公网访问”右侧的 ，关闭公网访问开关。

**步骤6** 单击 ，关闭公网访问功能。

关闭公网访问功能大约需要10~30秒，请耐心等待。关闭公网访问后，页面会自动跳转到“后台任务管理”页签，当任务状态为“成功”时，表示关闭公网访问成功。

----结束

## 6.8 设置实例镜像队列

镜像队列，允许集群将队列镜像到其他代理上，当集群某一代理宕机后，队列能自动切换到镜像中的其他代理，保证服务的可用性。

如果您需要了解RabbitMQ Web UI相关功能和概念，请自行查阅[RabbitMQ官网](#)。本章节仅介绍登录RabbitMQ实例的Web页面设置镜像队列的操作步骤。

### 操作步骤

**步骤1** 登录RabbitMQ实例的Web UI。

登录操作，请参考[连接RabbitMQ管理地址](#)。

**步骤2** 在菜单栏，选择“Admin”。

图 6-1 选择 Admin 菜单



**步骤3** (可选) 选择右侧导航栏“Virtual Hosts”，然后输入“Name”，单机“Add virtual host”，创建vhost。

如果您需要设置指定vhost，请执行本步骤；如果不需要，请直接执行**步骤4**。

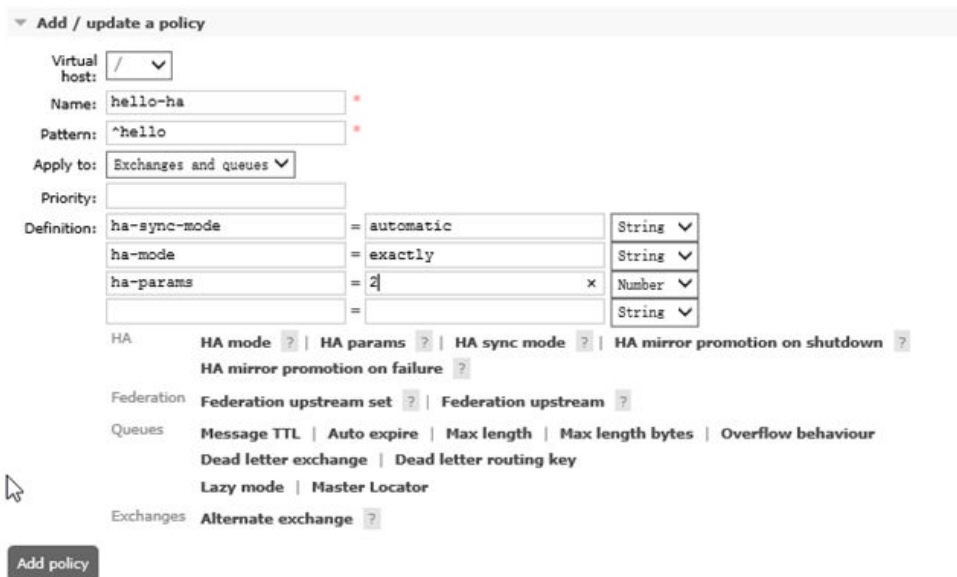
图 6-2 创建 vhost



**步骤4** 选择右侧导航栏“Policies”，为vhost设置规则。

如果为指定的vhost设置，请在“Virtual Host”选择**步骤3**创建的vhost；如果没有，则默认为“/”。

图 6-3 设置 vhost 规则



参数解释如下：

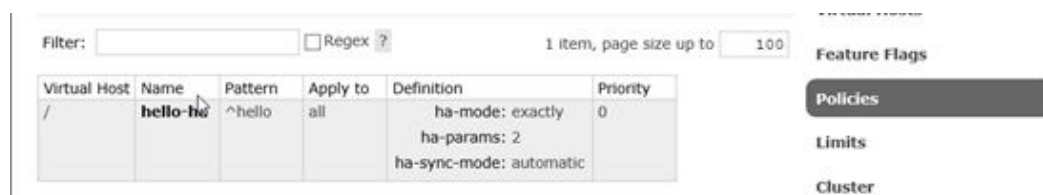
- Name: policy的名称，用户自定义。
- Pattern: queue的匹配模式（正则表达式）。

- Definition: 镜像定义，包括三个部分ha-sync-mode、ha-mode、ha-params。
  - ha-sync-mode: 表示镜像队列中消息的同步方式，有效取值范围为：automatic, manually。
    - automatic: 表示自动向master同步数据。
    - manually: 表示手动向master同步数据。
  - ha-mode: 指明镜像队列的模式，有效取值范围为all/exactly/nodes。
    - all: 表示在集群所有的代理上进行镜像。
    - exactly: 表示在指定个数的代理上进行镜像，代理的个数由ha-params指定。
    - nodes: 表示在指定的代理上进行镜像，代理名称通过ha-params指定。
  - ha-params: ha-mode模式需要用到的参数。
- Priority: 可选参数， policy的优先级。

步骤5 单击“Add policy”。

规则添加成功后如下图所示。

图 6-4 vhost 规则



----结束

## 6.9 开启实例插件

RabbitMQ实例创建后，支持开启如下插件，实例创建后默认都是关闭状态。

### 📖 说明

开启插件过程中，不会重启实例，但是以下插件（rabbitmq\_mqtt、rabbitmq\_web\_mqtt、rabbitmq\_stomp、rabbitmq\_web\_stomp）会重启keepalived，导致连接断开。连接断开后，是否会自动重连依赖于用户自身的业务逻辑。

表 6-3 支持修改状态的实例插件

插件名称	功能描述	端口号
rabbitmq_amqp1_0	表示实例是否支持AMQP1.0协议。	-
rabbitmq_delayed_message_exchange	表示实例是否开启消息延迟功能。插件延迟时间存在1%左右的误差，可能提前或者推迟发送消息给消费者。	-


插件名称	功能描述	端口号
rabbitmq_federation	表示实例是否开启消息同步功能。	-
rabbitmq_sharding	表示实例是否开启消息分片功能。	-
rabbitmq_shovel	表示实例是否开启消息迁移功能。	-
rabbitmq_tracing	表示实例是否开启消息追踪功能。	-
rabbitmq_mqtt	表示实例是否支持MQTT协议（TCP方式）。	1883
rabbitmq_web_mqtt	表示实例是否支持MQTT协议（WebSocket方式）。	15675
rabbitmq_stomp	表示实例是否支持STOMP协议（TCP方式）。	61613
rabbitmq_web_stomp	表示实例是否支持STOMP协议（WebSocket方式）。	15674
rabbitmq_consistent_hash_exchange	表示实例是否支持x-consistent-hash。	-

#### 📖 说明

插件端口号不支持修改。


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

#### 📖 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击待开启插件的实例名称，进入实例详情页面。

**步骤5** 在“插件管理”页签，单击待开启插件后的“开启”。

确认开启后，等待实例插件开启成功。

----结束

## 6.10 管理实例标签

标签是RabbitMQ实例的标识，为RabbitMQ实例添加标签，可以方便用户识别和管理拥有的RabbitMQ实例资源。

您可以在创建RabbitMQ实例时添加标签，也可以在RabbitMQ实例创建完成后，在实例的详情页添加标签，您最多可以给实例添加20个标签。另外，您还可以进行修改和删除标签。


标签共由两部分组成：“标签键”和“标签值”，其中，“标签键”和“标签值”的命名规则如表6-4所示。

表 6-4 标签命名规则

参数名称	规则
标签键	<ul style="list-style-type: none"> <li>不能为空。</li> <li>对于同一个实例，Key值唯一。</li> <li>长度不超过36个字符。</li> <li>不能包含“=”，“*”，“&lt;”，“&gt;”，“\”，“，”，“ ”，“/”。</li> <li>首尾字符不能为空格。</li> </ul>
标签值	<ul style="list-style-type: none"> <li>长度不超过43个字符。</li> <li>不能包含“=”，“*”，“&lt;”，“&gt;”，“\”，“，”，“ ”，“/”。</li> <li>首尾字符不能为空格。</li> </ul>


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 单击待设置标签的实例名称，进入实例详情页面。

**步骤5** 单击“标签”页签，进入标签管理页面。

界面显示该实例的标签列表。

**步骤6** 您可以根据实际需要，执行以下操作：

- 添加标签
  - 单击“添加标签”，弹出“添加标签”对话框。  
如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键值对。另外，您可以单击的“查看预定义标签”，系统会跳转到标签管理服务页面，查看已经预定义的标签，或者创建新的标签。  
您也可以直接在“标签键”和“标签值”中输入设置标签。
  - 单击“确定”。为实例添加标签成功。
- 修改标签

单击标签所在行“操作”列下的“编辑”，在弹出的“编辑标签”窗口，输入修改后标签的值，并单击“确定”。

- 删除标签

单击标签所在行“操作”列下的“删除”，如果确认删除，在弹出的“删除标签”窗口，单击“确定”。

----结束

## 6.11 删除队列

本章节指导您通过RabbitMQ WebUI页面或调用API方式删除队列。

- **方法一：在WebUI页面删除单个队列：**在WebUI页面的“Queues”页签中，删除单个队列。
- **方法二：通过Policy批量删除队列：**新增与待删除队列的前缀名称相同、且队列过期时间（TTL）为1毫秒的策略，通过此策略实现批量删除队列。
- **方法三：调用API删除单个队列：**在RabbitMQ实例未开启SSL时，通过调用API删除单个队列。
- **方法四：调用API批量删除队列：**在RabbitMQ实例未开启SSL时，通过编写Shell脚本循环调用API执行删除命令，实现批量删除队列。

### 须知

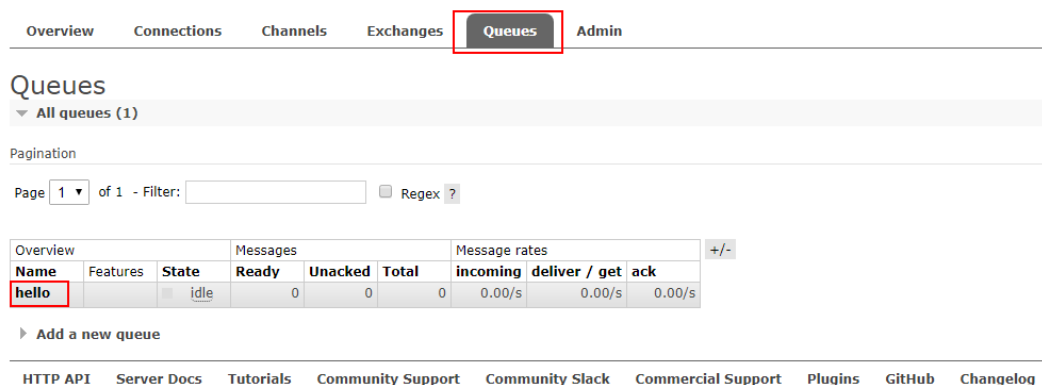
删除队列前，请确保队列中的消息已经全部被消费，否则未消费的消息将和队列一起被删除。

### 方法一：在 WebUI 页面删除单个队列

**步骤1** 登录RabbitMQ WebUI页面。

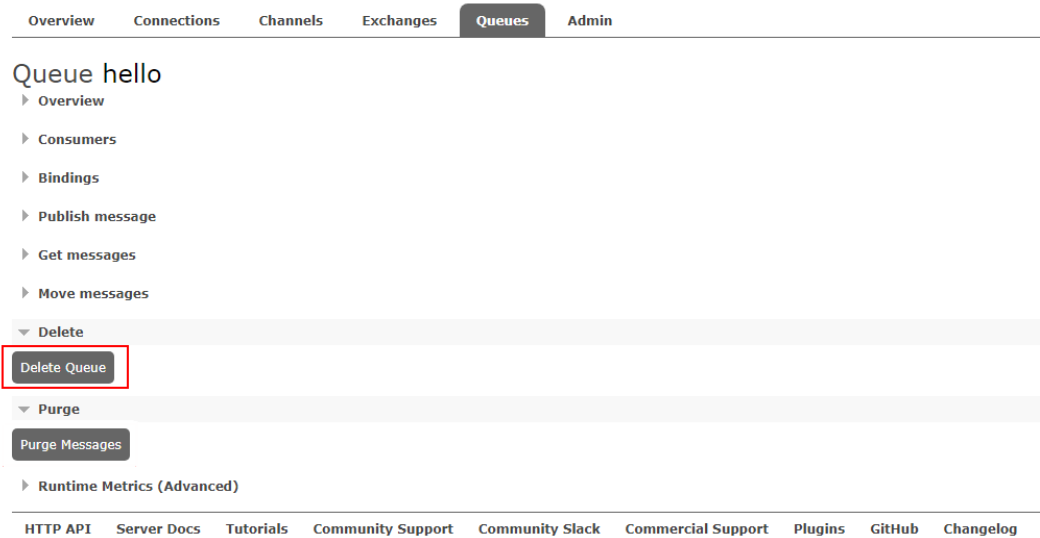
**步骤2** 在“Queues”页签，单击需要删除的队列名称，进入队列详情页面。

图 6-5 队列列表



**步骤3** 单击“Delete Queue”，删除单个队列。

图 6-6 删除单个队列



----结束

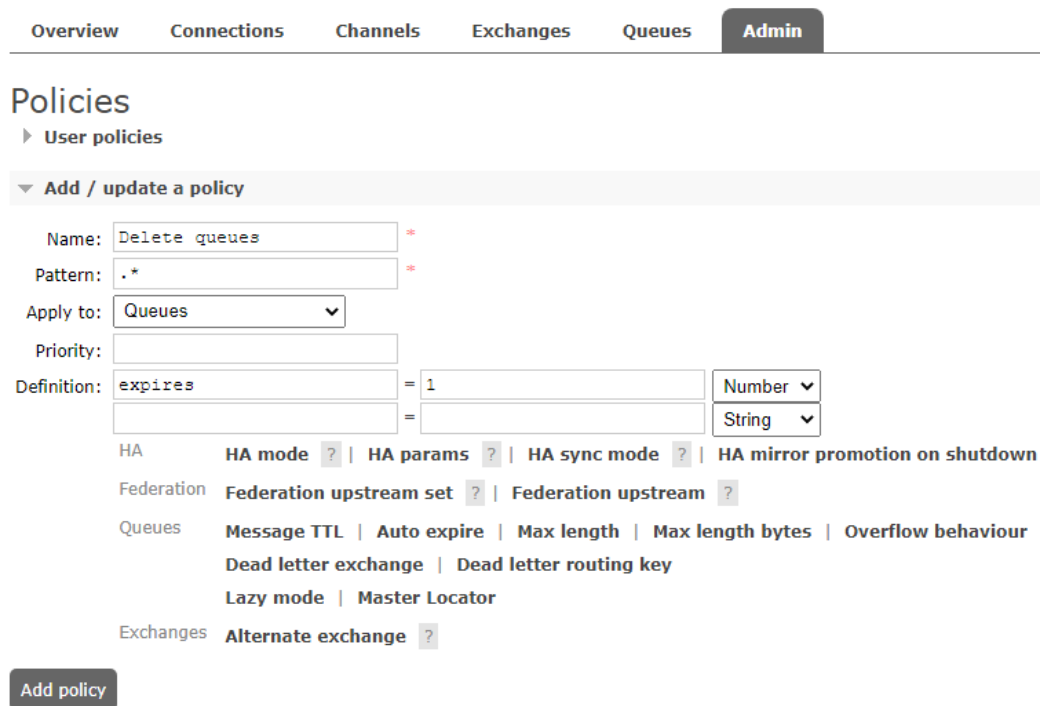
## 方法二：通过 Policy 批量删除队列

新增与待删除队列的前缀名称相同、且队列TTL为1毫秒的策略，通过此策略实现批量删除队列。

步骤1 [登录RabbitMQ WebUI页面](#)。

步骤2 在“Admin > Policies”页面中，新增一条策略。

图 6-7 通过 Policy 批量删除队列



- Name: 填写策略名称。
- Pattern: 队列匹配模式, 填写队列名称, 会匹配前缀同名的队列。例如: 设置为“.”时, 表示匹配所有队列。设置为“.\*queue-name”时, 表示匹配队列名前缀为queue-name的所有队列。
- Apply to: 选择“Queues”。
- Priority: 可选参数, 策略优先级, 数字越大, 优先级越高。
- Definition: 定义TTL, 单位为毫秒。填写“expires”参数, 值设置为“1”, 表示队列过期时间为1毫秒。

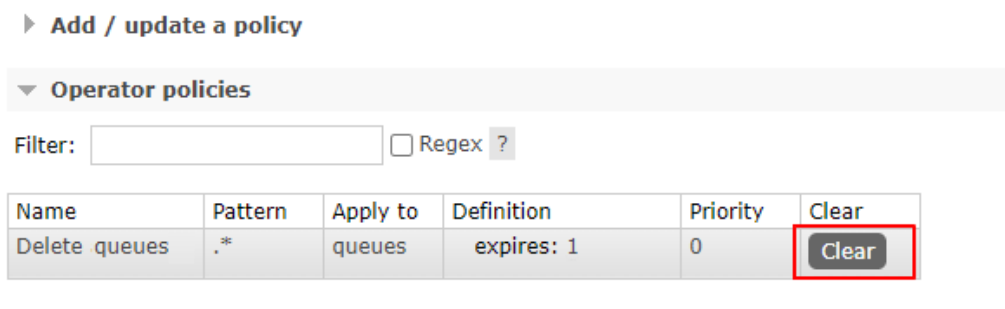
**步骤3** 单击“Add policy”。

在“Queues”页签, 查看队列是否成功删除。

**步骤4** 队列成功删除后, 在“Admin > Policies”页面中, 在**步骤2**中新增的策略后, 单击“Clear”, 删除策略。

如果保留此策略, 它对后续新建的队列依然生效, 可能会出现误删除队列的情况。

**图 6-8** 删除策略



----结束

### 方法三: 调用 API 删除单个队列

在RabbitMQ实例未开启SSL时, 通过调用API删除单个队列。

**步骤1** 在Linux系统中连接RabbitMQ实例, 具体步骤请参考[连接未开启SSL方式的RabbitMQ实例](#)。

**步骤2** 执行以下命令删除单个队列。

```
curl -i -XDELETE http://${USERNAME}:${PASSWORD}@${HOST}:${PORT}/api/queues/${VHOST_NAME}/${QUEUE_NAME}
```

参数说明如下:

- USERNAME: 创建实例时设置的用户名。
- PASSWORD: 创建实例时设置的密码, 如果忘记密码, 参考[重置实例密码](#), 重新设置密码。
- HOST: 在实例详情页, 查看Web界面UI地址。
- PORT: 在实例详情页, 查看Web界面UI端口号。
- VHOST\_NAME: vhost名称, 默认为“/”, 在命令中设置为“%2F”。
- QUEUE\_NAME: 待删除队列的名称。

示例如下:



```
curl -i -XDELETE http://test:Zsxxdx@192.168.0.241:15672/api/queues/%2F/hello
```

删除成功后，回显如下：

图 6-9 删除队列成功

```
HTTP/1.1 204 No Content
content-security-policy: default-src 'self'
date: Tue, 14 Jun 2022 02:52:23 GMT
server: Cowboy
vary: accept, accept-encoding, origin
```

您还可以在WebUI页面的“Queues”页签，查看队列是否成功删除。

----结束

## 方法四：调用 API 批量删除队列

在RabbitMQ实例未开启SSL时，通过编写Shell脚本循环调用API执行删除命令，实现批量删除队列。

**步骤1** 在Linux系统中连接RabbitMQ实例，具体步骤请参考[连接未开启SSL方式的RabbitMQ实例](#)。

**步骤2** 创建“delete\_queues.sh”脚本文件。

```
touch delete_queues.sh
```

**步骤3** 执行以下命令，编辑脚本。

```
vim delete_queues.sh
```

将以下内容复制到脚本中，其中USERNAME、PASSWORD、HOST和QUEUES\_LIST的值，请根据实际情况修改。

```
#!/usr/bin/env bash

USERNAME=root
PASSWORD=Zsxxdx
HOST=192.168.0.241
PORT=15672
VHOST=%2F

QUEUES_LIST="test1 test2 test3";
for QUEUE_NAME in $QUEUES_LIST :
do
    curl -i -XDELETE http://$USERNAME:$PASSWORD@$HOST:$PORT/api/queues/$VHOST/$QUEUE_NAME
done
```

参数说明如下：

- USERNAME：创建实例时设置的用户名。
- PASSWORD：创建实例时设置的密码，如果忘记密码，参考[重置实例密码](#)，重新设置密码。
- HOST：在实例详情页，查看Web界面UI地址。
- PORT：在实例详情页，查看Web界面UI端口号。
- VHOST：vhost名称，默认为“/”，在命令中设置为“%2F”。
- QUEUES\_LIST：待删除队列的名称，队列名称之间使用空格隔开。

**步骤4** 保存脚本内容。

**步骤5** 对脚本进行授权。

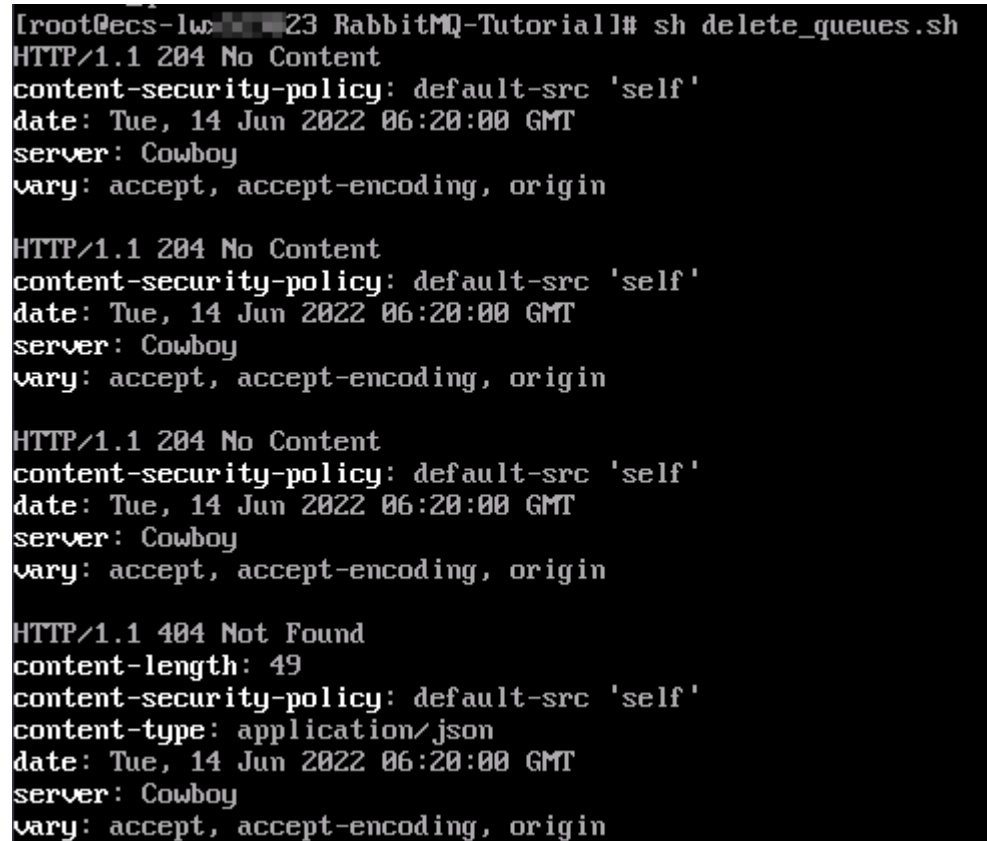
```
chmod 777 delete_queues.sh
```

**步骤6** 执行脚本。

```
sh delete_queues.sh
```

删除成功后，回显如下：

图 6-10 批量删除队列成功



```
[root@ecs-lw-23 RabbitMQ-Tutorial]# sh delete_queues.sh
HTTP/1.1 204 No Content
content-security-policy: default-src 'self'
date: Tue, 14 Jun 2022 06:20:00 GMT
server: Cowboy
vary: accept, accept-encoding, origin

HTTP/1.1 204 No Content
content-security-policy: default-src 'self'
date: Tue, 14 Jun 2022 06:20:00 GMT
server: Cowboy
vary: accept, accept-encoding, origin

HTTP/1.1 204 No Content
content-security-policy: default-src 'self'
date: Tue, 14 Jun 2022 06:20:00 GMT
server: Cowboy
vary: accept, accept-encoding, origin

HTTP/1.1 404 Not Found
content-length: 49
content-security-policy: default-src 'self'
content-type: application/json
date: Tue, 14 Jun 2022 06:20:00 GMT
server: Cowboy
vary: accept, accept-encoding, origin
```

您还可以在WebUI页面的“Queues”页签，查看队列是否成功删除。

----结束



# 7 调整资源配额

## 什么是配额？

为防止资源滥用，平台限定了各服务资源的配额，对用户的资源数量和容量做了限制。如您最多可以创建多少个RabbitMQ实例。

如果当前资源配额限制无法满足使用需要，您可以申请扩大配额。

## 怎样查看我的配额？

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择区域和项目。
3. 单击页面右上角的“My Quota”图标 。  
系统进入“服务配额”页面。
4. 您可以在“服务配额”页面，查看各项资源的总配额及使用情况。  
如果当前配额不能满足业务要求，请参考后续操作，申请扩大配额。

## 如何申请扩大配额？

目前系统暂不支持在线调整配额大小。如您需要调整配额，请拨打热线或发送邮件至客服，客服会及时为您处理配额调整的需求，并以电话或邮件的形式告知您实时进展。

在拨打热线或发送邮件之前，请您准备好以下信息：

- Account Name、Project Name、Project ID。  
获取方式如下：登录云帐户管理控制台，在右上角单击帐户名，选择“我的认证”，在“我的认证”页面获取相关信息。
- 配额信息，包括：
  - 服务名
  - 配额类别
  - 需要的配额值

如您需要调整配额，请联系管理员。

# 8 监控

## 8.1 支持的监控指标

### 功能说明

本节定义了分布式消息服务RabbitMQ版上报云监控服务的监控指标的命名空间，监控指标列表和维度定义，用户可以通过云监控服务提供管理控制台来检索DMS服务产生的监控指标和告警信息。

### 命名空间

SYS.DMS

### 实例监控指标

表 8-1 实例支持的监控指标

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
connections	连接数	该指标用于统计RabbitMQ实例中的总连接数。 单位: Count	>= 0	RabbitMQ实例	1分钟
channels	通道数	该指标用于统计RabbitMQ实例中的总通道数。 单位: Count	0~2047	RabbitMQ实例	1分钟
queues	队列数	该指标用于统计RabbitMQ实例中的总队列数。 单位: Count	0~1200	RabbitMQ实例	1分钟

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
consumers	消费者数	该指标用于统计 RabbitMQ实例中的总消费者数。 单位: Count	0~1200	RabbitMQ实例	1分钟
messages_ready	可消费消息数	该指标用于统计 RabbitMQ实例中总可消费消息数量。 单位: Count	0~100000	RabbitMQ实例	1分钟
messages_unacknowledged	未确认消息数	该指标用于统计 RabbitMQ实例中总已经消费但还未确认的消息数量。 单位: Count	0~100000	RabbitMQ实例	1分钟
publish	生产速率	统计RabbitMQ实例中实时消息生产速率。 单位: Count/s	0~25000	RabbitMQ实例	1分钟
deliver	消费速率(手工确认)	统计RabbitMQ实例中实时消息消费速率(手工确认)。 单位: Count/s	0~25000	RabbitMQ实例	1分钟
deliver_noack	消费速率(自动确认)	统计RabbitMQ实例中实时消息消费速率(自动确认)。 单位: Count/s	0~50000	RabbitMQ实例	1分钟

## 节点监控指标

表 8-2 节点支持的监控指标

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
fd_used	文件句柄数	该指标用于统计当前节点RabbitMQ所占用的文件句柄数。 单位: Count	0~65535	RabbitMQ实例节点	1分钟

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
socket_used	Socket连接数	该指标用于统计当前节点RabbitMQ所使用的Socket连接数。 单位: Count	0~50000	RabbitMQ实例节点	1分钟
proc_used	Erlang进程数	该指标用于统计当前节点RabbitMQ所使用的Erlang进程数。 单位: Count	0~1048576	RabbitMQ实例节点	1分钟
mem_used	内存占用	该指标用于统计当前节点RabbitMQ内存占用。 单位: Byte	0~3200000000	RabbitMQ实例节点	1分钟
disk_free	可用存储空间	该指标用于统计当前节点可使用的存储空间。 单位: Byte	0~5000000000	RabbitMQ实例节点	1分钟
rabbitmq_alive	节点存活状态	表示Rabbitmq节点是否存活。	1: 存活 0: 离线	RabbitMQ实例节点	1分钟
rabbitmq_disk_usage	磁盘容量使用率	统计Rabbitmq节点虚拟机的磁盘容量使用率。 单位: %	0~100%	RabbitMQ实例节点	1分钟
rabbitmq_cpu_usage	CPU利用率	统计Rabbitmq节点虚拟机的CPU使用率。 单位: %	0~100%	RabbitMQ实例节点	1分钟
rabbitmq_cpu_core_load	CPU核均负载	统计Rabbitmq节点虚拟机CPU每个核的平均负载。	>0	RabbitMQ实例节点	1分钟
rabbitmq_memory_usage	内存使用率	统计Rabbitmq节点虚拟机的内存使用率。 单位: %	0~100%	RabbitMQ实例节点	1分钟
rabbitmq_disk_read_await	磁盘平均读操作耗时	该指标用于统计磁盘在测量周期内平均每个读IO的操作时长。 单位: ms	>0	RabbitMQ实例节点	1分钟

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
rabbitmq_disk_write_wait	磁盘平均写操作耗时	该指标用于统计磁盘在测量周期内平均每个写IO的操作时长。 单位: ms	>0	RabbitMQ实例节点	1分钟
rabbitmq_node_bytes_in_rate	网络流入流量	统计Rabbitmq节点每秒网络访问流入流量。 单位: Byte/s	>0	RabbitMQ实例节点	1分钟
rabbitmq_node_bytes_out_rate	网络流出流量	统计Rabbitmq节点每秒网络访问流出流量。 单位: Byte/s	>0	RabbitMQ实例节点	1分钟
rabbitmq_node_queues	节点队列数	该指标用于统计Rabbitmq节点队列个数。 单位: 个	>0	RabbitMQ实例节点	1分钟
rabbitmq_memory_high_watermark	内存高水位状态	表示Rabbitmq节点是否触发内存高水位, 如果触发, 会阻塞集群的所有生产者。	1: 触发 0: 没有触发	RabbitMQ实例节点	1分钟
rabbitmq_disk_insufficient	磁盘高水位状态	表示Rabbitmq节点是否触发磁盘高水位, 如果触发, 会阻塞集群的所有生产者。	1: 触发 0: 没有触发	RabbitMQ实例节点	1分钟

## 队列监控指标

表 8-3 队列支持的监控指标

指标ID	指标名称	指标含义	取值范围	测量对象	监控周期 (原始指标)
queue_messages_unacknowledged	队列未确认消息数	该指标用于统计队列中已消费未确认消息数。 单位: Count	0~1000000	RabbitMQ实例队列	1分钟
queue_messages_ready	队列可消费消息数	该指标用于统计队列中可消费的消息数。 单位: Count	0~1000000	RabbitMQ实例队列	1分钟

## 维度

Key	Value
rabbitmq_instance_id	RabbitMQ实例
rabbitmq_node	RabbitMQ实例节点
rabbitmq_queue	RabbitMQ实例队列

## 8.2 设置 RabbitMQ 告警规则

本章节主要介绍部分监控指标的告警策略，以及配置操作。在实际业务中，建议按照以下告警策略，配置监控指标的告警规则。

表 8-4 RabbitMQ 实例配置告警的指标

指标名称	告警策略	指标说明	解决方案
内存高水位状态	告警阈值：原始值 >=1 连续触发次数：1 告警级别：致命	告警阈值为1表示触发内存高水位，会阻塞消息生产	<ul style="list-style-type: none"> <li>加快消费</li> <li>采用生产者确认的发送模式，并监控生产端消息生产速度和时长，当消息生产时长有明显增加时进行流控措施</li> </ul>
磁盘高水位状态	告警阈值：原始值 >=1 连续触发次数：1 告警级别：致命	告警阈值为1表示触发磁盘高水位，会阻塞消息生产	<ul style="list-style-type: none"> <li>减少惰性队列的消息堆积</li> <li>减少持久化队列的消息堆积</li> <li>删除队列</li> </ul>
内存使用率	告警阈值：原始值 >业务预期使用率（推荐30%） 连续触发次数：连续3~5个周期 告警级别：重要	该指标需要分别为每个节点设置内存使用率告警，避免触发内存高水位阻塞生产	<ul style="list-style-type: none"> <li>加快消费</li> <li>采用生产者确认的发送模式，并监控生产端消息生产速度和时长，当消息生产时长有明显增加时进行流控措施</li> </ul>
CPU使用率	告警阈值：原始值 >业务预期使用率（推荐70%） 连续触发次数：连续3~5个周期 告警级别：重要	该指标需要分别为每个节点设置CPU使用率告警，CPU使用率过高可能会影响生产速度	<ul style="list-style-type: none"> <li>减少镜像队列个数</li> <li>对于集群实例，建议扩容节点个数，然后进行节点间重平衡</li> </ul>




指标名称	告警策略	指标说明	解决方案
可消费消息数	告警阈值：原始值 >业务预期可消费消息数 连续触发次数：1 告警级别：重要	可消费消息数过多表示消息堆积	请参考 <a href="#">消息堆积的解决办法</a>
未确认消息数	告警阈值：原始值 >业务预期未确认消息数 连续触发次数：1 告警级别：重要	未确认消息数过多可能会导致消息堆积	<ul style="list-style-type: none"> <li>检查消费者是否异常</li> <li>检查消费者逻辑是否消耗时间过长</li> </ul>
连接数	告警阈值：原始值 >业务预期连接数 连续触发次数：1 告警级别：重要	连接数突增可能是流量变大的预警	需检查业务是否正常，可参考其他告警
通道数	告警阈值：原始值 >业务预期通道数 连续触发次数：1 告警级别：重要	通道数突增可能是流量变大的预警	需检查业务是否正常，可参考其他告警
Erlang进程数	告警阈值：原始值 >业务预期进程数 连续触发次数：1 告警级别：重要	进程数突增可能是流量变大的预警	需检查业务是否正常，可参考其他告警

### 📖 说明

- 告警阈值请根据业务预期数设置。例如，业务预期使用率35%，则告警阈值设置35%。
- 连续触发次数和告警级别可根据业务逻辑自行调整。


## 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。


### 📖 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 在RabbitMQ实例名称后，单击“查看监控数据”，进入“云监控”页面。

**步骤5** 在实例监控指标页面中，找到需要创建告警的指标项，鼠标移动到指标区域，然后单

击指标右上角的 ，进入“创建告警规则”页面。

**步骤6** 在告警规则页面，设置告警信息。

创建告警规则操作，请查看《云监控服务 用户指南》。

1. 设置告警名称和告警的描述。
2. 设置告警策略和告警级别。

例如，在进行指标监控时，如果连续3个周期，连接数原始值超过设置的值，则产生告警，如果未及时处理，则每一天发送一次告警通知。

3. 设置“发送通知”开关。当开启时，设置告警生效时间、产生告警时通知的对象以及触发的条件。
4. 单击“立即创建”，等待创建告警规则成功。

----结束

## 8.3 查看监控数据

### 操作场景


云监控对分布式消息服务RabbitMQ版的运行状态进行日常监控，可以通过控制台直观的查看分布式消息服务RabbitMQ版各项监控指标。

### 前提条件

已创建RabbitMQ实例，且实例中有可消费的消息。


### 操作步骤

**步骤1** 登录管理控制台。

**步骤2** 在管理控制台左上角单击 ，选择区域。

#### 说明

此处请选择与您的应用服务相同的区域。

**步骤3** 在管理控制台左上角单击 ，选择“应用服务 > 分布式消息服务 RabbitMQ”，进入分布式消息服务RabbitMQ专享版页面。

**步骤4** 在RabbitMQ实例名称后，单击“查看监控数据”。

跳转到云监控页面，查看实例、节点和队列的监控数据，数据更新周期为1分钟。

----结束

# 9 云审计服务支持的关键操作

## 9.1 云审计服务支持的 DMS for RabbitMQ 操作列表

通过云审计服务，您可以记录与分布式消息服务RabbitMQ版相关的操作事件，便于日后的查询、审计和回溯。

表 9-1 云审计服务支持的 DMS for RabbitMQ 操作列表

操作名称	资源类型	事件名称
删除后台任务成功	rabbitmq	deleteDMSBackendJobSuccess
删除后台任务失败	rabbitmq	deleteDMSBackendJobFailure
创建DMS实例订单成功	rabbitmq	createDMSInstanceOrderSuccess
创建DMS实例订单失败	rabbitmq	createDMSInstanceOrderFailure
修改DMS实例订单成功	rabbitmq	modifyDMSInstanceOrderSuccess
修改DMS实例订单失败	rabbitmq	modifyDMSInstanceOrderFailure
扩容实例成功	rabbitmq	extendDMSInstanceSuccess
扩容实例失败	rabbitmq	extendDMSInstanceFailure
重置DMS实例密码成功	rabbitmq	resetDMSInstancePasswordSuccess
重置DMS实例密码失败	rabbitmq	resetDMSInstancePasswordFailure
删除创建失败的DMS实例成功	rabbitmq	deleteDMSCreateFailureInstanceSuccess
删除创建失败的DMS实例失败	rabbitmq	deleteDMSCreateFailureInstanceFailure
重启DMS实例成功	rabbitmq	restartDMSInstanceSuccess

操作名称	资源类型	事件名称
重启DMS实例失败	rabbitmq	restartDMSInstanceFailure
批量删除DMS实例成功	rabbitmq	batchDeleteDMSInstanceSuccess
批量删除DMS实例失败	rabbitmq	batchDeleteDMSInstanceFailure
批量重启DMS实例成功	rabbitmq	batchRestartDMSInstanceSuccess
批量重启DMS实例失败	rabbitmq	batchRestartDMSInstanceFailure
修改DMS实例信息成功	rabbitmq	modifyDMSInstanceInfoSuccess
修改DMS实例信息失败	rabbitmq	modifyDMSInstanceInfoFailure
批量删除DMS实例任务	rabbitmq	batchDeleteDMSInstanceTask
删除DMS实例任务成功	rabbitmq	deleteDMSInstanceTaskSuccess
删除DMS实例任务失败	rabbitmq	deleteDMSInstanceTaskFailure
创建DMS实例任务成功	rabbitmq	createDMSInstanceTaskSuccess
创建DMS实例任务失败	rabbitmq	createDMSInstanceTaskFailure
扩容DMS实例任务成功	rabbitmq	extendDMSInstanceTaskSuccess
扩容DMS实例任务失败	rabbitmq	extendDMSInstanceTaskFailure
重启DMS实例任务成功	rabbitmq	restartDMSInstanceTaskSuccess
重启DMS实例任务失败	rabbitmq	restartDMSInstanceTaskFailure
批量重启DMS实例任务成功	rabbitmq	batchRestartDMSInstanceTaskSuccess
批量重启DMS实例任务失败	rabbitmq	batchRestartDMSInstanceTaskFailure
修改DMS实例信息任务成功	rabbitmq	modifyDMSInstanceInfoTaskSuccess
修改DMS实例信息任务失败	rabbitmq	modifyDMSInstanceInfoTaskFailure

## 9.2 查看云审计日志

查看DMS for RabbitMQ云审计日志，请参考[查看追踪事件](#)。

# 10 常见问题

## 10.1 实例问题

### 10.1.1 RabbitMQ 使用的版本是多少？

服务端RabbitMQ的版本是3.7.17。

### 10.1.2 RabbitMQ 实例 SSL 连接的协议版本号是多少？

TLS v1.2版本。

### 10.1.3 创建实例时为什么无法查看子网和安全组等信息？

创建实例时，如果无法查看虚拟私有云、子网、安全组、弹性IP，可能原因是该用户无Server Administrator和VPC Administrator权限，增加权限的详细步骤请参考《统一身份认证服务 用户指南》。

### 10.1.4 重启 RabbitMQ 实例时，若其中一台 RabbitMQ 重启失败，会如何处理？

重启RabbitMQ实例时，不会重启实例所在虚拟机，仅重启RabbitMQ进程。

重启集群实例时，若其中一台RabbitMQ进程重启失败，则重启后实例状态依然为“运行中”，并提示“部分节点故障”。在每台虚拟机上都有RabbitMQ的守护进程，定时检查RabbitMQ进程是否存在，当进程不存在时会自动拉起RabbitMQ进程。

如果RabbitMQ实例异常持续超过1分钟，会上报告警。

### 10.1.5 RabbitMQ 集群实例如何均衡分发请求到每个虚拟机？

集群内部使用LVS做负载均衡，由LVS将请求均衡分发到每个虚拟机节点。

### 10.1.6 RabbitMQ 实例集群内部的队列是否有冗余备份？

队列是否做镜像（即冗余备份）取决于用户的需要，如果用户设置了镜像，会在集群中多个代理上存储队列的副本，当某个代理故障，集群会从其他正常的代理中选择一个代理，用来同步队列数据。

## 10.1.7 RabbitMQ 实例是否支持持久化，如何定时备份数据？

RabbitMQ支持消息数据持久化，可从客户端连接RabbitMQ并设置消息持久化，也可在RabbitMQ集群管理工具界面创建队列时设置消息持久化。

不支持客户自定义定时备份数据，或从界面触发备份数据。

## 10.1.8 RabbitMQ 实例开启 SSL 后，证书怎么获取？

RabbitMQ实例开启SSL后只做单向认证，不需要证书。

## 10.1.9 RabbitMQ 实例的 SSL 开关是否支持修改？

不支持动态修改，即如果实例创建时没有选择开启，创建完成之后，不支持修改，建议在实例创建时将开关打开。

## 10.1.10 RabbitMQ 实例是否支持扩容？

单机版本的RabbitMQ实例不支持扩大存储空间。

集群版本的RabbitMQ实例不支持扩大存储空间，只支持扩节点。

## 10.1.11 RabbitMQ 实例是否支持 MQTT 协议？

支持启用MQTT插件，MQTT插件信息，可参考<https://www.rabbitmq.com/mqtt.html#enabling-plugin>。

RabbitMQ实例购买后，支持的插件如下：

方括号中为空的表示还未安装，标记为[E\*]的插件是明确安装的，标记为[e\*]的插件是隐式安装的，也就是说，这些插件是作为其它的插件的依赖而进行安装的。

```
[ ] rabbitmq_amqp1_0          3.7.17
[ ] rabbitmq_auth_backend_cache 3.7.17
[ ] rabbitmq_auth_backend_http 3.7.17
[ ] rabbitmq_auth_backend_ldap 3.7.17
[ ] rabbitmq_auth_mechanism_ssl 3.7.17
[ ] rabbitmq_consistent_hash_exchange 3.7.17
[ ] rabbitmq_delayed_message_exchange 3.8.0
[ ] rabbitmq_event_exchange    3.7.17
[ ] rabbitmq_federation        3.7.17
[ ] rabbitmq_federation_management 3.7.17
[ ] rabbitmq_jms_topic_exchange 3.7.17
[E*] rabbitmq_management       3.7.17
[e*] rabbitmq_management_agent 3.7.17
[ ] rabbitmq_mqtt              3.7.17
[ ] rabbitmq_peer_discovery_aws 3.7.17
[ ] rabbitmq_peer_discovery_common 3.7.17
[ ] rabbitmq_peer_discovery_consul 3.7.17
[ ] rabbitmq_peer_discovery_etcd 3.7.17
[ ] rabbitmq_peer_discovery_k8s 3.7.17
[ ] rabbitmq_random_exchange   3.7.17
[ ] rabbitmq_recent_history_exchange 3.7.17
[ ] rabbitmq_sharding          3.7.17
[ ] rabbitmq_shovel            3.7.17
[ ] rabbitmq_shovel_management 3.7.17
[ ] rabbitmq_stomp             3.7.17
[E*] rabbitmq_top              3.7.17
[ ] rabbitmq_tracing          3.7.17
[ ] rabbitmq_trust_store       3.7.17
[e*] rabbitmq_web_dispatch     3.7.17
[ ] rabbitmq_web_mqtt         3.7.17
```

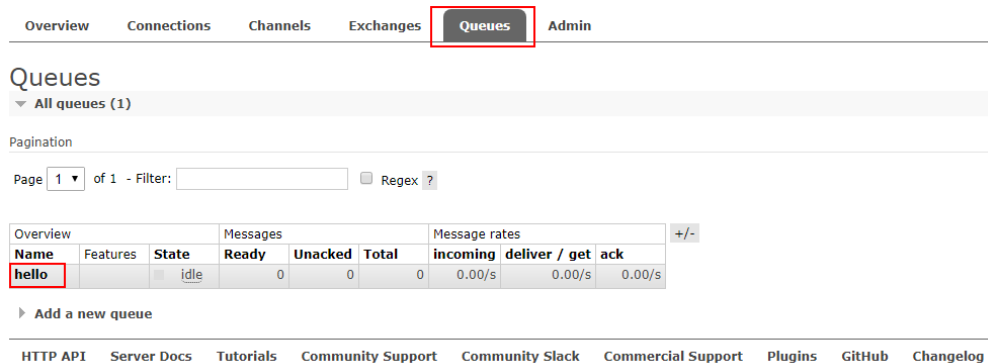
```
[ ] rabbitmq_web_mqtt_examples    3.7.17
[ ] rabbitmq_web_stomp           3.7.17
[ ] rabbitmq_web_stomp_examples  3.7.17
```

RabbitMQ控制台支持安装的插件有：rabbitmq\_amqp1\_0、rabbitmq\_delayed\_message\_exchange、rabbitmq\_federation、rabbitmq\_sharding、rabbitmq\_shovel、rabbitmq\_tracing、rabbitmq\_mqtt、rabbitmq\_web\_mqtt、rabbitmq\_stomp、rabbitmq\_web\_stomp和rabbitmq\_consistent\_hash\_exchange，实例创建后，以上插件默认关闭，如需开启，在控制台实例详情的“插件管理”页面开启。

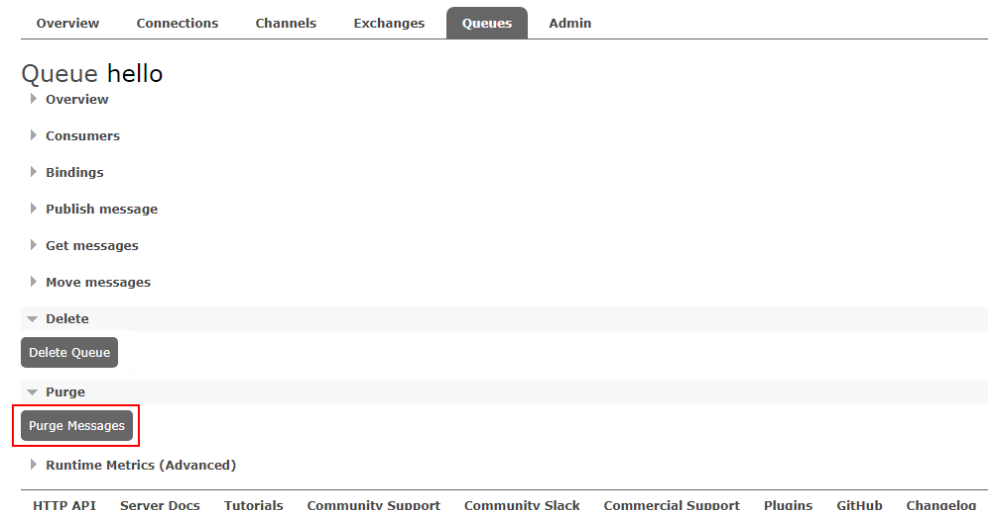
如果需要激活其他未安装且控制台不支持开启的插件（如rabbitmq\_random\_exchange），需要联系客服在后台开启插件，开启过程中，对业务没有任何影响。

## 10.1.12 如何清空队列数据？

1. 参考“连接RabbitMQ管理地址”章节，登录Web UI。
2. 在“Queues”页签，单击需要清空数据的队列名称，进入队列详情页面。



3. 单击“Purge Messages”，清空队列数据。



## 10.1.13 RabbitMQ 支持升级 CPU 和内存吗？

RabbitMQ不支持升级CPU和内存。

## 10.1.14 如何关闭 RabbitMQ 的 WebUI?

创建RabbitMQ实例后，如果想要关闭RabbitMQ的WebUI，只要您在安全组入方向中不开放15672端口（实例未开启SSL时的端口）或者15671（实例开启SSL时的端口），此时就无法登录WebUI界面。

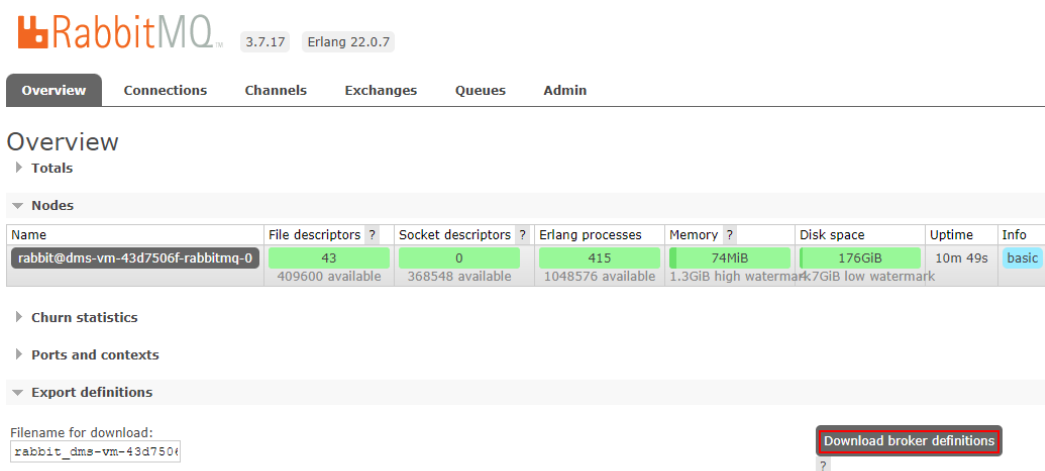
## 10.1.15 实例是否支持修改可用区?

不支持，您可以重新创建实例，以满足可用区要求，然后进行实例元数据的迁移。

实例元数据的迁移步骤如下：

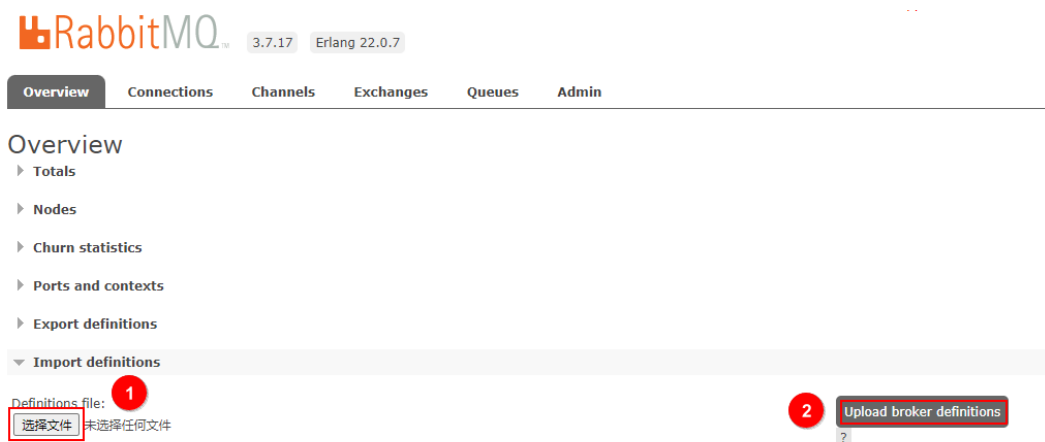
**步骤1** 登录重新创建前的RabbitMQ实例的WebUI页面。

**步骤2** 在“Overview”页签中，单击“Download broker definitions”，导出元数据。



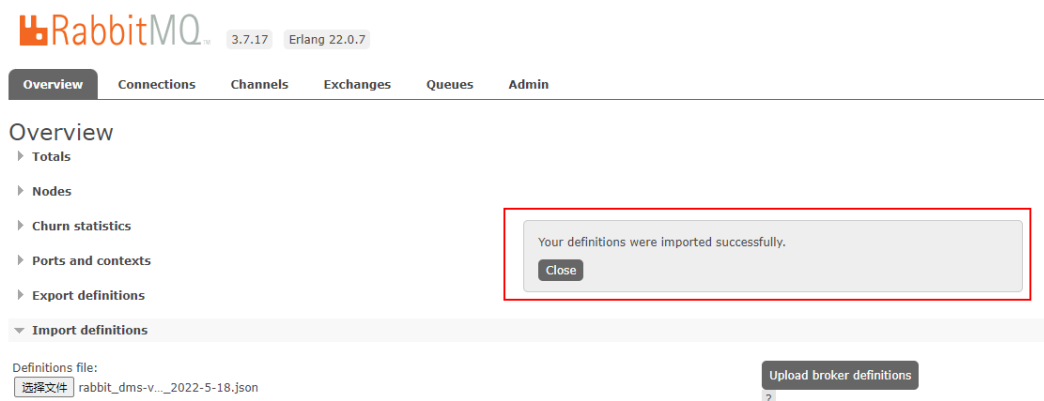
**步骤3** 登录重新创建的RabbitMQ实例的WebUI页面，在“Overview”页签中，单击“选择文件”，选择**步骤2**中导出的元数据。

**步骤4** 单击“Upload broker definitions”，上传元数据。



上传成功后，显示如下信息。





----结束

## 10.2 连接问题

### 10.2.1 如何配置安全组？

RabbitMQ实例支持VPC内访问和公网访问，配置安全组的方式如下：

- VPC内访问实例

客户端只能部署在与RabbitMQ实例/处于相同虚拟私有云（VPC）和相同子网的弹性云服务器（ECS）上。

除了ECS、RabbitMQ实例必须处于相同VPC和相同子网之外，还需要他们的安全组分别配置了正确的规则，客户端才能访问RabbitMQ实例。

- a. 建议ECS、RabbitMQ实例配置相同的安全组。安全组创建后，默认包含组内网络访问不受限制的规则。
- b. 如果配置了不同安全组，可参考如下配置方式：

#### 📖 说明

- 假设ECS、RabbitMQ实例分别配置了安全组：sg-53d4、sg-RabbitMQ、Default\_All。
- 以下规则，远端可使用安全组，也可以使用具体的IP地址。

ECS所在安全组需要增加如下规则，以保证客户端能正常访问RabbitMQ实例。

**表 10-1** 安全组规则

方向	协议端口	目的地址
出方向	全部放通	Default_All

RabbitMQ实例所在安全组需要增加如下规则，以保证能被客户端访问。

表 10-2 安全组规则

方向	协议端口	源地址
入方向	全部放通	sg-53d4

- 通过公网访问实例  
RabbitMQ实例安全组需要配置了正确的规则，客户端才能访问RabbitMQ实例。假设RabbitMQ实例安全组为sg-RabbitMQ，则需要配置如下入方向规则中的其中一种规则：
  - 协议为TCP，端口为5672，源IP为0.0.0.0/0。
  - 协议为Any，源IP为0.0.0.0/0。

## 10.2.2 RabbitMQ 客户端连接报错原因分析

RabbitMQ客户端连接失败，可能原因包括地址、端口填错、用户名或者密码填错、超过最大连接数。

- 连接地址不正确**

VPC内访问场景下，连接地址不正确时，报错如下：

```
[root@ecs-heru RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.125.110 5672 user
*****
Exception in thread "main" java.net.NoRouteToHostException: No route to host (Host unreachable)
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
```

公网访问场景下，连接地址不正确时，报错如下：

```
[root@ecs-heru RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 139.xxx.178 5672 user *****
Exception in thread "main" java.net.SocketTimeoutException: connect timed out
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
```

- 端口不正确**

VPC内访问场景下，端口不正确时，报错如下：

```
[root@ecs-heru RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.125.111 5673 user
*****
Exception in thread "main" java.net.ConnectException: Connection refused (Connection refused)
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
```

公网访问场景下，端口不正确时，报错如下：

```
[root@ecs-heru RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 139.xxx.179 5673 user *****
Exception in thread "main" java.net.SocketTimeoutException: connect timed out
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
```

- 用户名或密码错误**

```
[root@ecs-heru RabbitMQ-Tutorial]# java -cp ./rabbitmq-tutorial.jar Send 192.168.125.111 5672 user
*****
Exception in thread "main" com.rabbitmq.client.AuthenticationFailureException: ACCESS_REFUSED - Login was refused using authentication mechanism PLAIN. For details see the broker logfile.
at com.rabbitmq.client.impl.AMQConnection.start(AMQConnection.java:351)
at com.rabbitmq.client.impl.recovery.RecoveryAwareAMQConnectionFactory.newConnection(RecoveryAwareAMQConnectionFactory.java:64)
```

- 超过最大连接数

### 10.2.3 RabbitMQ 实例是否支持公网访问？

支持。在创建RabbitMQ实例时开启“公网访问”，或创建完后，在实例详情页中将公网访问开关打开。

### 10.2.4 RabbitMQ 是否支持跨 Region 部署？

当前支持跨AZ（可用区），不支持跨Region部署。

### 10.2.5 RabbitMQ 实例是否支持跨 VPC 和跨子网访问？

RabbitMQ实例支持跨VPC和子网访问，可以通过创建VPC对等连接，将两个VPC的网络打通，实现跨VPC访问实例。

### 10.2.6 RabbitMQ 实例是否支持不同的子网？

支持。

客户端与实例在相同VPC内，可以跨子网段访问。

客户端与实例在不同VPC时，需建立VPC对等连接。

此外，可以为实例绑定公网地址，客户端访问实例公网地址即可。

### 10.2.7 SSL 方式连接 RabbitMQ 实例失败？

首先排查安全组的入方向规则，是否放开了端口5671（SSL方式访问）或5672（非SSL访问）。

其次，参考如下内容配置SSL单向认证：

```
ConnectionFactory factory = new ConnectionFactory();
factory.setHost(host);
factory.setPort(port);
factory.setUsername(user);
factory.setPassword(password);
factory.useSslProtocol();
Connection connection = factory.newConnection();
Channel channel = connection.createChannel();
```

### 10.2.8 客户端是否可以通过 DNAT 方式访问 RabbitMQ 实例？

不可以。客户端可以使用代理、VPN、专线、FullNAT或者反向代理等方式访问RabbitMQ实例。

### 10.2.9 RabbitMQ 实例的 Web 管理页面无法打开

可能原因：实例安全组配置不正确

解决方案：重新配置安全组，具体步骤如下。

1. 在实例详情页面的“基本信息 > 网络”，单击安全组名称，跳转到安全组页面。
2. 选择“入方向规则”，查看安全组入方向规则。
  - 实例未开启SSL开关

- 如果是VPC内访问，实例安全组入方向规则，需要允许端口5672的访问。
- 如果是公网访问，需要允许端口15672的访问。
- 实例已开启SSL开关
  - 如果是VPC内访问，实例安全组入方向规则，需要允许端口5671的访问。
  - 如果是公网访问，需要运行端口15671的访问。

## 10.2.10 客户端是否可以连接同个 RabbitMQ 下多个 vhost?

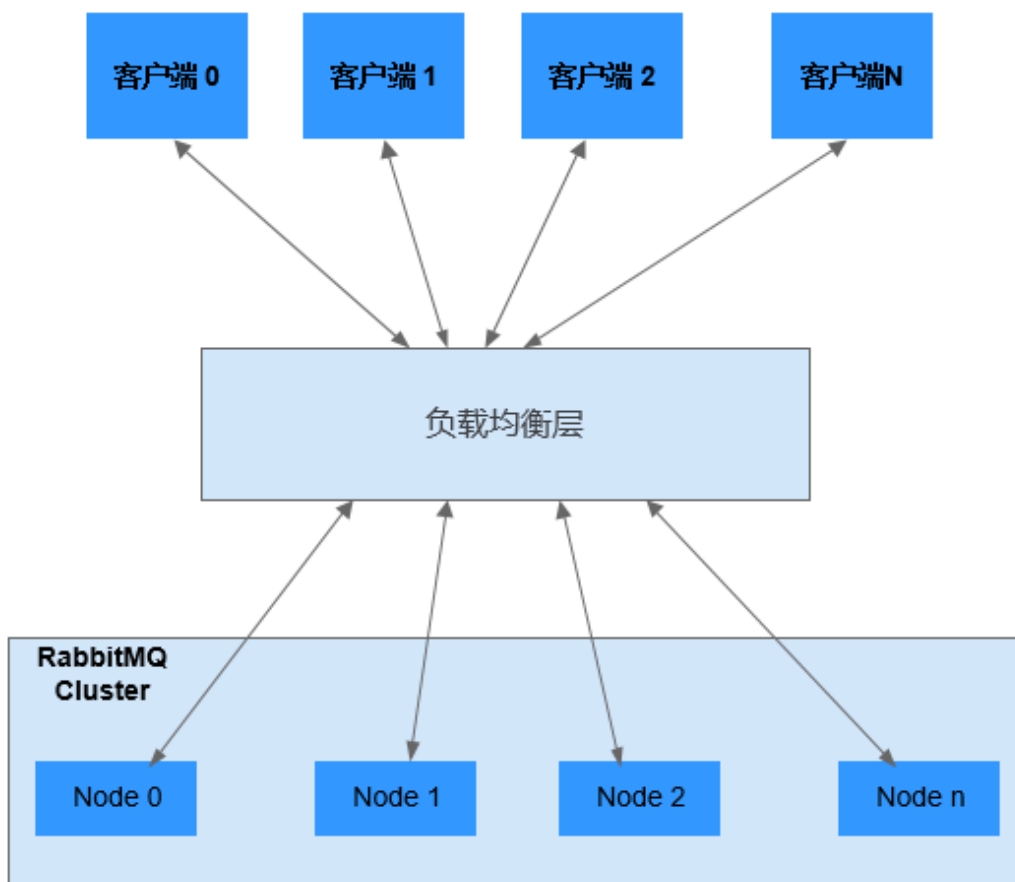
客户端可以连接同个RabbitMQ下多个vhost。

vhost ( Virtual Hosts ) 是RabbitMQ的基本特性，每个vhost相当于一个相对独立的RabbitMQ服务器，每个vhost数据目录不同，共用一个进程。性能上，连接多个vhost和单独使用一个vhost差别不大，只是RabbitMQ进程多一些对象，建议使用业务模型实测。

vhost的相关介绍，请参考官网文档[Virtual Hosts](#)。

## 10.2.11 为什么 RabbitMQ 集群只有一个连接地址?

RabbitMQ集群实例的连接地址，实际上是实例的LVS节点地址（负载均衡地址），客户端连接实例时，通过负载均衡器将客户端请求分发到集群实例的各个节点。



## 10.3 插件问题

### 10.3.1 支持的 RabbitMQ 插件有哪些？

#### 支持的 RabbitMQ 插件有哪些？

RabbitMQ实例购买后，支持的插件如下，其中，方括号中为空的表示还未安装，标记为[E\*]的插件是明确安装的，标记为[e\*]的插件是隐式安装的，也就是说，这些插件是作为其它的插件的依赖而进行安装的。

```
[ ] rabbitmq_amqp1_0          3.7.17
[ ] rabbitmq_auth_backend_cache 3.7.17
[ ] rabbitmq_auth_backend_http 3.7.17
[ ] rabbitmq_auth_backend_ldap 3.7.17
[ ] rabbitmq_auth_mechanism_ssl 3.7.17
[ ] rabbitmq_consistent_hash_exchange 3.7.17
[ ] rabbitmq_delayed_message_exchange 3.8.0
[ ] rabbitmq_event_exchange    3.7.17
[ ] rabbitmq_federation        3.7.17
[ ] rabbitmq_federation_management 3.7.17
[ ] rabbitmq_jms_topic_exchange 3.7.17
[E*] rabbitmq_management       3.7.17
[e*] rabbitmq_management_agent 3.7.17
[ ] rabbitmq_mqtt              3.7.17
[ ] rabbitmq_peer_discovery_aws 3.7.17
[ ] rabbitmq_peer_discovery_common 3.7.17
[ ] rabbitmq_peer_discovery_consul 3.7.17
[ ] rabbitmq_peer_discovery_etcd 3.7.17
[ ] rabbitmq_peer_discovery_k8s 3.7.17
[ ] rabbitmq_random_exchange   3.7.17
[ ] rabbitmq_recent_history_exchange 3.7.17
[ ] rabbitmq_sharding          3.7.17
[ ] rabbitmq_shovel           3.7.17
[ ] rabbitmq_shovel_management 3.7.17
[ ] rabbitmq_stomp            3.7.17
[E*] rabbitmq_top             3.7.17
[ ] rabbitmq_tracing          3.7.17
[ ] rabbitmq_trust_store      3.7.17
[e*] rabbitmq_web_dispatch    3.7.17
[ ] rabbitmq_web_mqtt         3.7.17
[ ] rabbitmq_web_mqtt_examples 3.7.17
[ ] rabbitmq_web_stomp        3.7.17
[ ] rabbitmq_web_stomp_examples 3.7.17
```

RabbitMQ控制台支持安装的插件有：rabbitmq\_amqp1\_0、rabbitmq\_delayed\_message\_exchange、rabbitmq\_federation、rabbitmq\_sharding、rabbitmq\_shovel、rabbitmq\_tracing、rabbitmq\_mqtt、rabbitmq\_web\_mqtt、rabbitmq\_stomp、rabbitmq\_web\_stomp和rabbitmq\_consistent\_hash\_exchange，实例创建后，以上插件默认关闭，如需开启，在控制台实例详情的“插件管理”页面开启。

如果需要激活其他未安装的插件，需要联系客服在后台开启插件，开启过程中，对业务没有任何影响。

## 10.4 消息问题

## 10.4.1 RabbitMQ 实例支持延时消息队列么？

RabbitMQ可以通过设置消息的有效期、和死信队列来实现延迟消息。同时，也提供安装插件实现延迟消息。当前RabbitMQ支持的插件：rabbitmq\_amqp1\_0、rabbitmq\_delayed\_message\_exchange、rabbitmq\_federation、rabbitmq\_sharding、rabbitmq\_shovel、rabbitmq\_tracing、rabbitmq\_mqtt、rabbitmq\_web\_mqtt、rabbitmq\_stomp、rabbitmq\_web\_stomp和rabbitmq\_consistent\_hash\_exchange。

## 10.4.2 消息堆积对业务的影响及解决办法

### 消息堆积对业务的影响

过量的消息堆积可能会引起内存或磁盘告警，从而造成所有connection阻塞，进而影响到其他队列的使用，导致整体服务质量的下降。

### 消息堆积产生的原因

1. 一般来说消息堆积是由于生产消息的速率远大于消费消息的速率所导致的。比如某个时间段消费端处理消息异常缓慢，发送一条消息只要3秒钟，而消费一条消息需要1分钟，每分钟发送20个消息，只有一个消息被消费端处理，这样队列中就会产生大量的消息堆积。
2. 消费者出现异常，生产者一直在发送消息，但是消费者不能消费，造成消息积压。
3. 消费者没有出现异常，但是消费者与队列间的订阅可能出现了异常，也会导致消息无法被消费从而造成堆积的情况。
4. 消费者正常，与队列间的订阅也正常，但是消费端的代码本身逻辑耗费时间长导致了消费能力降低，这时候就会出现1中的情况从而导致消息堆积。

### 解决消息堆积的办法

1. **生产速率较快，消费速率较慢**：您可以通过以下方法解决。
  - 增加消费者数量提高消费速率。
  - 采用生产者确认的发送模式，并监控生产端消息生产速度和时长，当消息生产时长有明显增加时进行流控措施。
2. **消费者异常**：建议排查消费者逻辑是不是有问题，优化程序。
3. **消费者与队列间的订阅异常**：建议排查队列和消费者之间的订阅是否正常。
4. **消费端的代码本身逻辑耗费时间长**：建议给消息设置过期时间，设置方法如下：
  - 在生产消息时，设置消息过期时间。消息过期时间以expiration值体现。

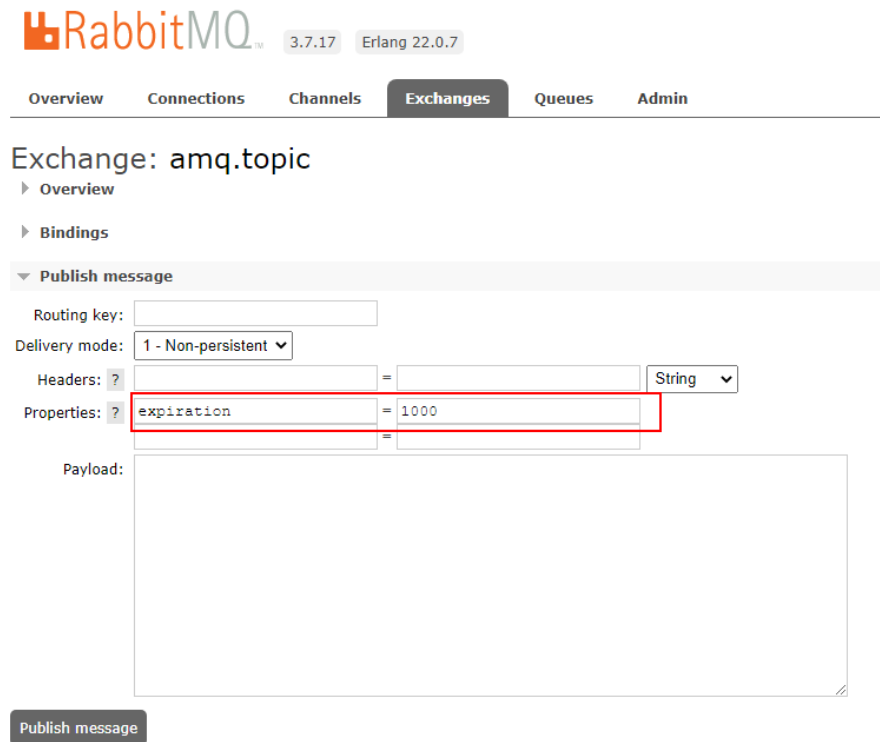
- 在properties中设置expiration值，单位为毫秒(ms)。

```
AMQP.BasicProperties properties = new AMQP.BasicProperties().builder()
    .deliveryMode(2)
    .contentEncoding("UTF-8")
    .expiration("10000")
    .build();
```

```
String message = "hello rabbitmq";
channel.basicPublish(exchange, routingKey, properties,
    message.getBytes(StandardCharsets.UTF_8));
```

- 在Web界面中设置expiration值，单位为毫秒(ms)。

[登录Web界面](#)，在“Exchanges”页签，单击Exchange名称，进入Exchange详情页。在“Publish message”区域，设置expiration值，如下图所示。



- 设置队列过期时间。队列过期时间以x-message-ttl值体现。从消息进入队列开始计算，超过了配置的队列过期时间，消息会自动被删除。
  - 在客户端代码中设置x-message-ttl值，单位为毫秒(ms)。

```
Map<String, Object> arguments = new HashMap<String, Object>();  
arguments.put("x-message-ttl", 10000);  
channel.queueDeclare(queueName, true, false, false, arguments);
```
  - 在Web界面新建队列时，设置x-message-ttl值，单位为毫秒(ms)。  
[登录Web界面](#)，在“Exchanges”页签，新建队列时，设置x-message-ttl值，如下图所示。

The screenshot shows the RabbitMQ Admin interface. At the top, there's a navigation bar with 'Overview', 'Connections', 'Channels', 'Exchanges', 'Queues', and 'Admin'. The 'Exchanges' tab is selected. Below the navigation bar, the page title is 'Exchanges' and it shows a list of 7 exchanges. The 'Add a new exchange' form is visible below the list, with the 'Arguments' field containing 'x-message-ttl = 10000' highlighted in red.

Name	Type	Features	Message rate in	Message rate out	+/-
(AMQP default)	direct	D			
amq.direct	direct	D			
amq.fanout	fanout	D			
amq.headers	headers	D			
amq.match	headers	D			
amq.rabbitmq.trace	topic	D I			
amq.topic	topic	D			

**Add a new exchange**

Name:

Type:

Durability:

Auto delete: ?

Internal: ?

Arguments:

Add **Alternate exchange** ?

### 10.4.3 消息的最长保留时间是多久？

一般情况下消息如果未被消费会一直保留，只有被消费后，才会被删除。但是如果设置了过期时间（TTL），则以TTL时间为准。

## 10.5 监控告警问题

### 10.5.1 云监控无法展示 RabbitMQ 监控数据

监控数据无法展示，可能原因：队列名称开头包含特殊字符，例如点号“.”、下划线“\_”，建议删除带特殊字符的队列。

### 10.5.2 云监控显示通道数一直上升报警有影响吗？

一个连接最大通道数是2047，超过后再创建通道数会失败，建议排查是否为资源没有释放导致的。



# A 修订记录

---

发布日期	修订记录
2022-08-12	第一次正式发布。