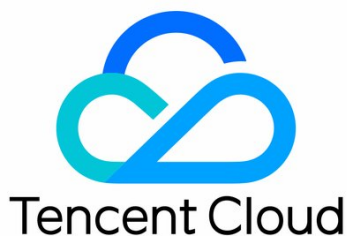


# 消息队列 CKafka

## 操作指南



#### Copyright Notice

©2013–2023 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

#### Trademark Notice

 Tencent Cloud

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

#### Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## 操作指南

### 实例管理

- 创建实例
- 批量连续命名或指定模式串命名
- 查看实例
- 升配实例
- 降配实例
- 销毁/退还实例
- 按量计费实例转包年包月
- 升级实例版本
- 添加路由策略
- 公网带宽管理
- 接入外部监控 Prometheus
- 迁移可用区
- 设置系统维护时间
- 配置消息大小

### Topic 管理

- 实例 Topic
  - 创建 Topic
  - 查看 Topic
  - 删除 Topic
  - 发送消息
  - 设置 Topic 限流规则
- 弹性 Topic
  - Topic 管理
  - 订阅管理

### Consumer Group

- 概述
- 创建 Consumer Group
- 查询 Consumer Group
- 删除 Consumer Group
- 设置 Offset
- 自动重置 offset

### 监控告警

- 查看监控信息
- 查询高级监控（专业版）
- 配置告警
- 告警配置建议
- 查看高级运维特性（专业版）

### 智能运维

- 一键诊断
- 弹性伸缩
  - 磁盘水位处理
  - 分区动态处理

### 权限管理

- 配置 ACL 策略
- 访问管理 CAM

### 标签管理

- 标签概述
- 使用标签
- 编辑标签

### 查询消息

事件中心

迁移上云

迁移方案概述

步骤1: 购买云上实例

步骤2: 迁移 Topic 上云

步骤3: 迁移数据上云

方案1: 单写双消费迁移

方案2: 单写单消费迁移

方案3: 使用 Mirrmaker 迁移

方案4: 迁移未消费数据

集群备份

数据压缩

# 操作指南

## 实例管理

### 创建实例

Last updated: 2023-05-24 16:59:04

#### 操作场景

该任务指导您通过 CKafka 控制台创建实例和 Topic，快速了解 CKafka 控制台操作流程。

#### 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击**新建**进入实例购买页。
3. 在实例购买页，根据自身业务需求选择购买信息。
  - 计费模式：专业版实例支持**包年包月**和**按量计费**两种模式，标准版实例支持**包年包月**模式。
  - 规格类型：根据自身业务需求选择标准版或者专业版。
  - Kafka 版本：根据您的业务需求选择 Kafka 版本，可参见 [CKafka 版本选择建议](#)。
  - 地域：选择和部署客户端的资源相近的地域。
  - 可用区：根据实际需要选择可用区。
    - 标准版：不支持多可用区部署。
    - 专业版：若当前地域支持多可用区部署，则最多可选择4个可用区进行部署。关于跨可用区部署原理介绍请参见 [跨可用区部署](#)。
  - 产品规格：根据峰值带宽和磁盘容量选择对应的型号。
  - 消息保留：范围在 24 ~ 2160 小时。  
在磁盘容量不足（即磁盘水位达到90%）时，将会提前删除旧的消息，以确保服务可用性。
  - 私有网络：若用户需要接入其他私有网络可参见 [添加路由策略](#) 修改路由接入规则。
  - 公网带宽：CKafka 默认赠送3Mbps公网带宽，专业版实例支持升配公网带宽，若您有更高的带宽需求，您可以额外支付费用购买。
  - 标签：选填，具体使用方法可参见 [标签管理](#)。
  - 实例名称：购买多个实例时，支持创建实例后缀数字自动升序以及指定模式串功能。具体操作参见 [批量连续命名或指定模式串命名](#)。
  - 自动续费：账户余额足够时，实例和公网带宽到期后自动按月续费。
4. 在鼠标放置在右下角的配置费用，可以查看到计费项明细。

配置费用明细			
计费项	原价	优惠	折扣价
磁盘	¥200.00	70.82%	¥56.64元
实例	¥100.00	-	¥100.00元

配置费用 ¥156.64 [立即购买](#)

5. 单击**立即购买**，完成实例创建。

ID/名称	监控	状态	可用区	实例类型	配置	网络类型	实例计费模式	公网计费模式	标签	操作
ckafka- 容灾		健康	广州六区 广州七区	专业版	Topic数量上 限: 450个 Partition数量上 限: 900个 版本: 2.4.1 磁盘类型: 高性能云硬盘 峰值带宽: 40 MB/s 磁盘容量: 500GB	私有网络	包年包月 2023-05-03到期	包年包月 2023-05-03到期	Application: underlay-domain a:555555	   

# 批量连续命名或指定模式串命名

Last updated: 2023-04-11 09:54:24

## 操作场景

在创建多个 CKafka 实例过程中，如果您希望实例名称具有一定的规则性，我们提供批量创建实例后缀数字自动升序功能以及指定模式串功能，您可以通过购买页和云 API 两种方式实现。

- **后缀数字自动升序**：如果购买多个的情况下设置实例名称，默认在设置名称后边加上数字的形式来标志多个 CKafka 实例（例如 ckafka1、ckafka2、ckafka3 ……），参见 [后缀数字自动升序](#)。
- **指定模式串**：
  - **指定单个模式串**：适用于需要创建 n 个实例并指定实例名称带有序号且序号从 x 开始递增时（例如 ckafka3、ckafka4、ckafka5 ……），参见 [指定单个模式串](#)。
  - **指定多个模式串**：适用于希望创建 n 个有多个前缀且每个前缀均指定序号的实例名称时（例如 ckafka3-big10-test、ckafka4-big11-test、ckafka5-big12-test ……），参见 [指定多个模式串](#)。

## 操作步骤

### 后缀数字自动升序

可将批量购买的实例设置为前缀相同，仅序号递增的实例名称。

#### ⚠ 注意

- 创建成功的实例默认序号从1开始递增，且不能指定开始的序号。
- 以下操作以您购买了3个 CKafka 实例，并希望生成的实例名称为“ckafka+序号”（即实例名称为 ckafka1、ckafka2 和 ckafka3）为例。

#### 购买页操作

1. 参见 [创建实例](#) 购买3个实例，并在购买页以“前缀+序号”的命名规则填写实例名称，即将实例名称填写为 ckafka 。如下图所示：



2. 根据页面提示逐步操作，并完成支付。

#### API 操作

在云 API [ModifyInstanceAttributes](#) 中，设置相关字段：

实例名称：将 InstanceName 字段指定为 ckafka 。

### 指定模式串

可将批量购买的实例设置为复杂且指定序号的实例名称。实例名称支持指定单个或者多个模式串，在设置实例名称时，请根据实际需求进行设置。指定模式串的命名：`**[R:x]**`，x 表示生成实例名称的初始序号，只支持正整数，不支持负数和浮点数。

#### 指定单个模式串

以下操作以您需要创建3个实例，且指定实例的序号从3开始递增为例。

#### 购买页操作

1. 参见 [创建实例](#) 购买实例，并在购买页以“前缀+指定模式串{R:x}”的命名规则填写实例名称，即将实例名称填写为 ckafka{R:3} 。如下图所示：

实例名称 ⓘ

ckafka{R:3}

支持批量连续命名或指定模式串命名，您还可以输入53个字符

2. 根据页面提示逐步操作，并完成支付。

## API 操作

在云 API [ModifyInstanceAttributes](#) 中，设置相关字段：

实例名称：将 InstanceName 字段指定为 ckafka{R:3}。

## 指定多个模式串

以下操作以您需要创建3个实例，并希望生成实例名称含有 ckafka、big 和 test 前缀，且 ckafka 和 big 前缀后面带序号，序号分别从13和2开始递增（即实例名称为 ckafka13-big2-test、ckafka14-big3-test、ckafka15-big4-test）为例。

## 购买页操作

1. 参见 [创建实例](#) 购买3台实例，并购买页以“前缀+指定模式串{R:x}-前缀+指定模式串{R:x}-前缀”的命名规则填写实例名称，即将实例名称填写为 ckafka{R:13}-big{R:2}-test。如下图所示：

实例名称 ⓘ

ckafka{R:13}-big{R:2}-test

支持批量连续命名或指定模式串命名，您还可以输入38个字符

2. 根据页面提示逐步操作，并完成支付。

## API 操作

在云 API [ModifyInstanceAttributes](#) 中，设置相关字段：

实例名称：将 InstanceName 字段指定为 ckafka{R:13}-big{R:2}-test。

## 验证功能

当您通过 [后缀数字自动升序](#) 或 [指定模式串](#) 实现批量创建实例后，可通过以下操作验证设置实例名称。

登录 [CKafka 控制台](#) 查看新创建的实例，即可发现批量购买的实例会根据您设置的规则进行命名。如下图所示：

ID/名称	监控	状态	可用区	实例类型	配置	网络类型	计费模式	标签	操作
<a href="#">ckafka-14</a> ckafka14-big3-test		健康	上海四区	标准版-入门型 版本: 1.1.1	峰值带宽: 40 MB/s 磁盘容量: 300GB	私有网络	包年包月 2021-08-20到期		<a href="#">配置告警</a> <a href="#">续费</a> <a href="#">更多</a>
<a href="#">ckafka-ya</a> ckafka13-big2-test		健康	上海四区	标准版-入门型 版本: 1.1.1	峰值带宽: 40 MB/s 磁盘容量: 300GB	私有网络	包年包月 2021-08-20到期		<a href="#">配置告警</a> <a href="#">续费</a> <a href="#">更多</a>

# 查看实例

Last updated: 2023-04-11 09:54:24

## 操作场景

该任务指导您通过 CKafka 控制台查看实例的配置信息和健康状态。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，在**基本信息**页，可查看实例的健康状态、配置信息、接入方式、消息保留和自动创建 Topic 等信息。



### 说明

- 配置信息中内网 IP 与端口（例如 10.6.206.110:9092），表示用于获取后端服务的通讯地址，真实访问地址中端口可能存在多个，如果您的服务器配置了访问限制，请在服务器上放通9092 - 9192端口（broker 可能会自动扩容，扩容后需要放通的端口会增加，需要预留充足的数量）。
- 如果您开启了自动创建 Topic，将会在服务器上启用主题的自动创建，使用或获取不存在的主题元数据时，将自动使用配置的副本数和分区数进行创建。
- 自动创建的 Topic，总数量会根据实例的不同规格有不同的限制。详情请参见 [计费概述](#)。

## 健康状态说明

CKafka 对每个实例均设置有巡检程序，巡检程序会检查该实例的连接数、磁盘使用百分比、生产峰值带宽、消费峰值带宽，当这些指标超过一定的阈值后会产生不同的健康状态。说明如下：

指标	阈值 (N)	状态描述
连接数 (默认最大值5000)	$N \leq 80\%$	健康
	$80\% < N \leq 95\%$	告警
	$N > 95\%$	异常
磁盘使用百分比	$N \leq 80\%$	健康
	$80\% < N \leq 95\%$	告警
	$N > 95\%$	异常
生产峰值带宽 (不含副本带宽)	$N \leq 80\%$	健康
	$80\% < N \leq 100\%$	告警
	$N > 100\%$	异常
消费峰值带宽	$N \leq 80\%$	健康
	$80\% < N \leq 100\%$	告警
	$N > 100\%$	异常

**注意**

连接数默认最大值是5000，阈值判断是基于最大值的百分比进行判断。实例连接超过该最大值会导致客户端无法创建新的连接，如评估该最大值在实际业务中不合理可以 [提交工单](#) 申请扩大。

## 升配实例

Last updated: 2023-05-24 15:10:24

## 操作场景

如果当前的实例规格不能满足您的业务需求，可以通过控制台提升您的实例规格。

### 说明

- 通过控制台直接提升实例规格包括峰值带宽、磁盘容量、Partition 个数。实例规格的升配操作为平滑升级，您的服务不会中断。
- 升配可能会新增端口，可以稍后在实例详情页 - 接入方式 - 查看所有IP和端口进行查看。**配置安全组时，需要放开所有端口。**

## 前提条件

升配前请您进行如下检查：

- 检查实例是否存在不可用的公网路由，支撑网，VPC 网络等。参见 [添加路由策略](#)。
- 检查实例是否存在有未同步的副本。参见 [查看 Topic 详情](#)。
- 检查实例是否存在未完成的任务（数据迁移），是否存在创建异常的 Topic、删除异常的 Topic 数据等。

### 说明

如果存在上述未完成事项及任务，建议等待全部完成后再进行升配，如果其中任务存在执行异常情况，请联系腾讯云 [在线客服](#) 处理。

## 操作步骤

### 场景一：升级标准版实例规格

- 登录 [CKafka 控制台](#)。
- 在实例列表页的操作栏，选择**更多 > 升配**进入升配页。
- 在实例升配页，选择目标升配规格。

**升级配置**

原配置	型号	峰值带宽 (MB/s)	磁盘容量 (GB)	Topic上限 (个)	Partition上限 (个)
	入门型	40	500	25	60

目标规格类型:  标准版  专业版

目标实例类型:  入门型  标准型  进阶型

峰值带宽: 150MB/s

磁盘容量:  1000 4000 - 1000 + GB

Topic上限: 50个

Partition上限: 150个

升配可能会新增端口，可以稍后在实例详情页 - 接入方式 - 查看所有IP和端口进行查看。

总计费用①: ■ ■ ■

- 单击**提交**，根据页面提示，完成实例升配。

### 场景二：标准版升级至专业版

- 登录 [CKafka 控制台](#)。
- 在实例列表页的操作栏，选择**更多 > 升配**进入升配页。
- 在实例升配页，目标规格类型选择**专业版**。

### 升级配置

原配置	型号	峰值带宽 (MB/s)	磁盘容量 (GB)	Topic上限 (个)	Partition上限 (个)
	入门型	40	500	25	60

目标规格类型: 标准版 专业版

目标实例规格: 40~1200MB/s 1200MB/s以上

峰值带宽:  40 - 40 + MB/s

磁盘容量:  500 - 500 + GB

Topic上限: 最高可用Topic数量为400

Partition上限: - 800 +

Rebalance时间:  立刻执行  自定义时间 (可选择未来24小时内的任意时间)  
 2022-11-02 11:51:29

变配模式:  稳定模式 (预计耗时0小时35分钟)  
CKafka将限制变配过程中数据迁移速度, 最大程度保留实例的带宽性能, 适合于不希望干扰业务的场景  
 高速模式 (预计耗时0小时10分钟)  
CKafka将不对变配过程中数据迁移的速度进行限制, 会影响实例的生产消费带宽, 适合于业务低峰或者允许停服的场景

本次升配需要底层资源变更, 需要进行数据迁移, 数据迁移过程中, 当每个分区迁移完成后, 会进行分区的Leader切换, Leader切换的风险请[查看文档](#)

升配可能会新增端口, 可以稍后在实例详情页 - 接入方式 - 查看所有IP和端口进行查看。

总计费用① 📄

提交
关闭

- 产品规格: 根据峰值带宽和磁盘容量选择对应的型号。
- 实例价格: 升配按天补足差价, 具体请参见 [包年包月与产品配置升级说明](#)。
- rebalance时间:
  - 当识别到升配需要进行数据迁移时, 可以选择立即执行或者自定义时间 (推荐选择夜晚执行, 减少对业务的影响), 预计耗时由后台接口经过对变配升配的计算后得出。

Rebalance时间  立刻执行  自定义时间 (可选择未来24小时内的任意时间)  

2021-03-16 23:30:08

  
 预计耗时: 4小时44分钟

- 当识别到升配不需要进行数据迁移时, 则显示“本次升配不会产生Reblance”。

Rebalance时间: 本次升配不会产生Rebalance

- 升配模式: 当识别到升配需要进行实例迁移时, 可以根据实际业务需要选择升配模式; 若不需要进行实例迁移, 则无需选择升配模式。
  - 稳定模式: CKafka 将限制升配过程中数据迁移速度, 最大程度保留实例的带宽属性, 适合于不希望干扰业务的场景。
  - 高速模式: CKafka 将不对升配过程中数据迁移的速度进行限制, 会影响实例的生产消费带宽, 适合于业务低峰或者允许停服的场景。

4. 单击提交完成实例升配, 在状态列可实时查看实例的升配进度。

5. 若设置了定时升配, 则在状态栏可修改定时时间。

### 修改Rebalance时间 ×

Rebalance时间  📅

预计耗时 **4小时44分钟**  
预计于2021-03-17 04:14:34完成

提交
关闭

### 场景三：升级专业版实例规格

1. 登录 [CKafka 控制台](#)。
2. 在实例列表页的操作栏，选择**更多** > **升配**进入升配页。
3. 在实例升配页，选择目标升配规格。

### 升级配置 ×

原配置	型号	峰值带宽 (MB/s)	磁盘容量 (GB)	Partition上限 (个)
	40~1200MB/s	40	500	800

目标实例规格 40~1200MB/s 1200MB/s以上

峰值带宽  - 480 + MB/s

磁盘容量  - 3000 + GB

Topic上限 **最高可用Topic数量为1000**

Partition上限 - 2000 +

Rebalance时间  立刻执行  
 自定义时间 (可选择未来24小时内的任意时间)  
 📅

变配模式  稳定模式 (预计耗时0小时35分钟)  
CKafka将限制变配过程中数据迁移速度，最大程度保留实例的带宽性能，适合于不希望干扰业务的场景  
 高速模式 (预计耗时0小时10分钟)  
CKafka将不对变配过程中数据迁移的速度进行限制，会影响实例的生产消费带宽，适合于业务低峰或者允许停服升级的场景

本次升配需要底层资源变更，需要进行数据迁移。数据迁移过程中，当每个分区迁移完成后，会进行分区的Leader切换，Leader切换的风险请[查看文档](#)

升配可能会新增端口，可以稍后在实例详情页 - 接入方式 - 查看所有IP和端口进行查看。

总计费用 ①

提交
关闭

- **产品规格**：根据峰值带宽和磁盘容量选择对应的型号。
- **实例价格**：升配按天补足差价，具体请参见 [包年包月与产品配置升级说明](#)。
- **rebalance 时间**：
  - 当识别到升配需要进行数据迁移时，可以选择立即执行或者自定义时间（推荐选择夜晚执行，减少对业务的影响），预计耗时由后台接口经过对变配升配的计算后得出。

Rebalance时间  立刻执行  自定义时间 (可选择未来24小时内的任意时间)

📅

预计耗时 **4小时44分钟**

- 当识别到升配不需要进行数据迁移时，则显示“本次升配不会产生Rebalance”。

Rebalance时间 本次升配不会产生Rebalance

- 升配模式：当识别到升配需要进行实例迁移时，可以根据实际业务需要选择升配模式；若不需要进行实例迁移，则无需选择升配模式。
  - 稳定模式：CKafka 将限制升配过程中数据迁移速度，最大程度保留实例的带宽属性，适合于不希望干扰业务的场景。
  - 高速模式：CKafka 将不对升配过程中数据迁移的速度进行限制，会影响实例的生产消费带款，适合于业务低峰或者允许停服的场景。

4. 单击提交完成实例升配，在状态列可实时查看实例的升配进度。

ID/名称	监控	状态	可用区	实例类型	配置	网络类型	实例计费模式	公网计费模式	标签	操作
ckafka-		变配中 进度: 1%	圣保罗一区	标准版-入门型 版本: 1.1.1 磁盘类型: 高性能云硬盘	Topic数量上限: 25个 Partition数量上限: 60个 峰值带宽: 40 MB/s 磁盘容量: 500GB(含额外购买200GB)	私有网络	包年包月	2023-04-16到期		配置告警 续费 更多

5. 若设置了定时升配，则在状态栏可修改定时时间。

### 修改升级时间

升级时间

## 升配失败的可能原因

1. 当前可用区的磁盘资源不满足此次升配的需求，建议联系腾讯云客服确认是否有足够的资源。
2. 实例升配过程中如果选择高速模式，并且集群当中存在占用带宽资源较高的生产任务时，会发生数据迁移延迟时增长大，可通过 [查看监控](#) 观察升配时间段生产和消费流量是否存在过高峰值。
3. 升配过程中耗时过长，由于迁移的机器配置接受最大的消息字节数是1MB，而需要迁移的 broker 配置的是8MB。会导致 broker 无法接收超大消息迁移，从而产生较长的迁移数据耗时，请咨询腾讯云 [在线客服](#) 处理。
4. 新旧集群升配或迁移过程中，broker ip更新发生异常，导致新集群的 broker ip 拉取数据失败。通过 [查看监控](#) 可观察到持续一段时间无监控数据，请咨询腾讯云 [在线客服](#) 处理。

# 降配实例

Last updated: 2023-05-24 16:17:20

## 操作场景

如果当前的实例规格资源冗余较多，可以通过控制台降低您的专业版实例规格。

### 说明：

- 为保证集群稳定性，每个实例每周只能降配3次。如需要额外放开限制，您可以 [提交工单](#) 申请。
- 通过控制台直接降低实例规格包括峰值带宽、磁盘容量、Partition 个数。实例规格的降配操作为平滑变更，您的服务不会中断。
- 标准版实例不支持降低配置。

## 前提条件

降配前请您进行如下检查：

1. 检查实例是否存在不可用的公网路由，支撑网，VPC 网络等。参见 [添加路由策略](#)。
2. 检查实例是否存在有未同步的副本。参见 [查看 Topic 详情](#)。
3. 检查实例是否存在未完成任务（数据迁移），是否存在创建异常的 Topic、删除异常的 Topic 数据等。
4. 检查实例至少7天内的资源使用情况。

### 说明

如果存在上述未完成事项及任务，建议等待全部完成后再进行降配，如果其中任务存在执行异常情况，请联系腾讯云 [在线客服](#) 处理。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在实例列表页的操作栏，选择[更多](#) > [降配](#)进入升配页。
3. 在实例降配页，选择目标降配规格。

### 降低配置 ✕

原配置

型号	峰值带宽 (MB/s)	磁盘容量 (GB)	Partition上限 (个)
20~1200MB/s	40	500	

使用情况 (近7天)

生产带宽均值 (M...	消费带宽均值 (M...	已使用磁盘均值 (...	Topic数量 (个)	Partition数量 (个)
0/320	0/320	0/500	2	4

近7天峰值带宽 **0MB/s**

近7天峰值存储 **0GB**

峰值带宽  20 40  20  MB/s

磁盘容量  200 500  500  GB

Topic数量 最高可用topic数量为400

Partition数量  800    
您当前已使用Partition为4，为保证业务运行，无法将Partition降低到小于当前使用Partition数量。

Rebalance时间  立刻执行   
 自定义时间 (可选择未来24小时内的任意时间)

变配模式  稳定模式 (预计耗时0小时35分钟)   
CKafka将限制变配过程中数据迁移速度，最大程度保留实例的带宽性能，适合于不希望干扰业务的场景   
 高速模式 (预计耗时0小时10分钟)   
CKafka将对变配过程中数据迁移的速度进行限制，会影响实例的生产消费带宽，适合于业务低峰或者允许停服升级的场景

本次变配需要底层资源变更，需要进行数据迁移。数据迁移过程中，当每个分区迁移完成后，会进行分区的Leader切换，Leader切换的风险请[查看文档](#)

总计退费① 元

**说明:**

若近7天峰值带宽或近7天峰值存储大于目标规格，则不建议进行降配操作。

4. 点击**提交**，根据页面提示，完成实例降配。

# 销毁/退还实例

Last updated: 2023-07-06 09:29:32

## 操作场景

用户不再需要消息队列 CKafka 实例时，可以销毁并释放该实例。

消息队列 CKafka 实例的生命周期是指实例从启动到释放所经历的状态。通过对实例从启动到销毁期间的合理的管理，可确保运行于实例上的应用程序能高效经济地提供服务。实例有以下状态：

状态名	状态属性	状态描述
创建中	中间状态	实例创建后，进入运行中之前的状态。
正在运行	稳定状态	实例正常运行状态，表明您的磁盘、流量、连接数都处于规划范围内。
删除中	中间状态	实例受控制台或通过 API 执行删除操作。
已隔离	中间状态	实例受控制台或通过 API 执行销毁操作或您的实例已经欠费了，进入7天隔离状态。
创建失败	中间状态	实例受控制台或通过 API 执行购买操作扣费成功但分配实例失败，如遇到这种情况请联系 <a href="#">在线客服</a> 处理。
删除失败	稳定状态	实例被手动删除或者在到期14天后（包括第14天）未进行续费，CKafka 执行资源释放时失败。

## 操作步骤

### 手动销毁

针对未到期的包年包月实例，您可以选择手动销毁。操作步骤如下：

1. 登录 [CKafka 控制台](#)。
2. 在实例列表页的操作栏，选择**更多 > 销毁/退还**。
3. 在确认销毁的弹窗中，单击**提交**，即可销毁该实例。



### 注意

- 包年包月实例销毁后，在 CKafka 控制台保留7天，状态显示为已隔离。
- 处于已隔离状态的实例不能生产和消费数据，已保存在 CKafka 内的数据以及配置不会被销毁，**到期的消息仍然会依照开源 Kafka 机制被自动删除**。
- 对于7天内已隔离状态的实例，可以在控制台实例列表页，单击操作列的**续费**。续费成功后，实例即可恢复正在运行状态，实例可正常使用。
- 费用退还说明，请参见 [退费说明](#)。

### 到期/欠费自动销毁

包年包月类型实例到期/欠费后，最多在 CKafka 控制台中保留7个自然日。7天内完成续费可继续使用，详情请查看 [欠费说明](#)。

若您的 CKafka 实例在到期7天后（包括第7天）未进行续费，系统将在到期后约24小时内对资源释放，到期实例中的数据将被清除且不可恢复。

## 删除实例

包年包月实例在被手动销毁或者到期/欠费自动销毁后，会在 CKafka 控制台保留7天，状态显示为已隔离，7天后（包括第7天）彻底释放，在此期间您可以选择直接彻底删除该实例。

### 注意

删除后所有数据将被清除且不可恢复，请提前备份数据。

1. 登录 [CKafka 控制台](#)。
2. 在实例列表页的操作栏，选择**更多** > **销毁/退还**。
3. 在确认删除的弹窗中，单击**提交**，即可删除该实例。



# 按量计费实例转包年包月

Last updated: 2023-05-24 14:42:20

## 操作场景

为了方便您使用，CKafka 开放了按量计费实例转换成包年包月实例的功能，将临时使用的按量计费实例转换成长期并稳定使用的包年包月实例。您可以在 CKafka 控制台进行转换操作。

## 转换规则

在 CKafka 控制台提供计费模式转换功能，具体规则如下：

- 按量计费实例转包年包月时会产生一个续费订单，必须完成该订单的支付流程，计费方式的变更才能生效。若未支付或未成功支付，该订单可在您的 [订单中心](#) 页面查看和处理。
- 计费模式由按量计费转换为包年包月的实例不支持五天内无理由退还。
- 计费方式转换成功及支付成功后，实例会即刻按照包年包月计费，新包年包月实例的起始时间为转换成功时间。
- 在未成功支付前，不可对该实例进行重复计费模式转换。
- 在未成功支付前，实例配置信息发生变化（例如调整配置、重装系统、调整带宽和调整磁盘等），新购订单金额和实例不匹配，未支付订单会被禁止支付，您需要在 [订单中心](#) 取消当前未支付订单，再执行新的转换操作。
- 按量计费转包年包月功能支持同步转换实例和磁盘的计费方式。**实例计费模式转换与公网带宽计费模式转换无关。**
- 按量计费实例转换成包年包月后，包年包月的实例无法转换回按量计费。
- 当前不支持批量转换。

## 操作步骤

- 登录 [CKafka 控制台](#)。
- 在实例列表页的操作栏，选择按量计费实例，选择**更多** > 按量计费转包年包月。
- 在弹出的**按量计费转包年包月**窗口中，根据实际需求，设置续费时长以及是否自动续费。

### 按量计费转包年包月

按量计费实例转换成包年包月后，包年包月的实例无法转换回按量计费。  
公网带宽计费模式不会随实例计费模式转换。

购买时长

自动续费  账户余额足够时，设备到期后按月自动续费

总计费用  元

已阅读并同意[按量计费转包年包月规则](#)

- 单击**确认**，根据页面提示，完成转换操作。

# 升级实例版本

Last updated: 2023-07-25 16:03:11

消息队列 CKafka 专业版支持进行开源版本升级，帮助用户升级实例至更新版本，从而体验更丰富的功能。同时支持升级内核小版本，实现新功能使用、性能提升、问题修复等功能。

## 操作场景

### ● 自动升级触发场景（内核小版本）

- 场景一：当 CKafka 版出现重大 bug 或者安全漏洞时，系统会在您的维护时间内发起数据库内核小版本的升级，并提前通过站内信、短信等方式推送升级通知。
- 场景二：当 CKafka 版发生触发集群迁移的操作（如升配集群规格、扩缩容磁盘容量、CKafka 版本升级等）时，系统会将您的集群升级至最新内核小版本。

### ● 手动升级场景（内核小版本+开源 Kafka 版本）

除自动升级场景外，用户可自主在控制台手动升级内核小版本和开源 Kafka 版本。

## 注意事项

- 仅专业版支持。
- 版本升级完成时会涉及集群切换（即秒级实例连接断开），建议程序有自动重连功能，并且建议选择实例可维护时间内做切换。
- 支持低版本实例向高版本升级，不支持降级兼容版本。
- 实例的小版本升级，系统会自动检测，不支持自定义目标版本。
- 实例兼容版本升级，不涉及费用变更。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在**基本信息**模块，单击实例版本右侧的**升级版本**，设置好目标升级版本和升级时间。

### 升级版本

类型  升级内核小版本  升级kafka版本

当前版本 2.4.1

目标版本 2.8.1  
开源kafka版本之间的差异，可[参考文档](#)

执行时间  立即执行  自定义时间（可选未来24小时内的任意时间）

### ○ 升级内核小版本：

- 当前版本：当前的内核版本。
- 目标版本：默认升级到最新 Broker 版本，系统会自动检测小版本，如果小版本**升级**按钮无法选择，表示该实例已经是最新的小版本。内核小版本之间的差异，可查看 [参考文档](#)。
- 执行时间：可以选择**立即执行**或者**自定义时间**（可选未来24小时内的任意时间），建议选择业务低峰期进行。

### ○ 升级 Kafka 版本：

- 当前版本：当前 Kafka 版本。
- 目标版本：选择要升级的目标 Kafka 版本。开源 Kafka 版本之间的差异，可查看 [参考文档](#)。
- 执行时间：可以选择**立即执行**或者**自定义时间**（可选未来24小时内的任意时间），建议选择业务低峰期进行。

### 📌 说明：

当前 kafka 版本升级不支持1.1.1版本升级到2.4.1版本。

- 单击**确定**，提交升级任务。
- 在页面上方选择**事件中心**页签，可以看到一条升级实例版本的记录。



事件名称	状态	事件类型	影响对象	最近更新时间	操作
kafka版本升级	初始化	Broker版本变更	-	2022-06-14 14:48:05	<a href="#">查看详情</a> <a href="#">配置告警</a>

共 1 条 20 条 / 页

- 单击该记录操作列的**查看详情**，可以查看详细的升级任务进展。



事件详情

创建时间	2022-06-14 14:48:05
最近更新时间	2022-06-14 14:48:05
原配置	2.4.1
目标配置	2.8.1
进度	50 % 进度 2 / 4项

- **任务初始化**
- **服务器变更**  
进度 0%
- **配置变更**
- **版本信息变更**

- 回到实例列表页面，可以看到实例的状态变为**变配中**，同时可以看到升级进度。

## 添加路由策略

Last updated: 2023-04-11 09:54:24

### 操作场景

该任务指导您在使用消息队列 CKafka 时，通过控制台配置路由接入规则，增强对公网/内网传输中的用户网络访问控制。关于公网的访问请参见 [配置 ACL 策略](#)。

路由类型	VPC 网络	公网域名接入
接入方式	<ul style="list-style-type: none"> <li>PLAINTEXT</li> <li>SASL_PLAINTEXT</li> <li>SASL_SSL (仅专业版实例支持)</li> <li>SASL_SCRAM_256 (仅 2.4.1 和 2.8.1 版本实例支持，存量实例需要 <a href="#">提交工单</a> 申请)</li> <li>SASL_SCRAM_512 (仅 2.4.1 和 2.8.1 版本实例支持，存量实例需要 <a href="#">提交工单</a> 申请)</li> </ul>	<ul style="list-style-type: none"> <li>SASL_PLAINTEXT</li> <li>SASL_SSL (仅专业版实例支持)</li> </ul>

### 操作步骤

#### 说明

一个实例最多可以创建5条路由，其中公网路由有且仅可以有一条。

#### VPC 网络

**操作场景：**您购买实例时选择私有网络并选择了相应的 VPC 环境（例如 VPC A），表示仅能所选择的 VPC A 访问您的消息队列 CKafka 服务（生产数据、消费数据等）；若后续使用过程中发现其他 VPC 环境（例如 VPC B）有需求访问 VPC A 内的消息队列 CKafka 服务，则可以通过配置接入方式，选择 VPC 网络的路由策略。

**操作建议：**为了保证安全性，该接入方式提供 ACL 策略配置，以管理用户访问权限，请酌情配置。

**操作步骤：**

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击接入方式模块中的**添加路由策略**。
4. 在弹窗中，路由类型选择 **VPC 网络**，选择好接入方式和网络。

#### 添加路由策略

路由类型

接入方式

网络

如果现有的网络不合适，您可以去控制台[新建私有网络](#)或[新建子网](#)

IP

如果没有指定IP,系统会自动分配

提交

关闭

#### 说明

选择 VPC 网络接入时，支持指定 IP，当变更接入方式时可以通过指定 IP 来保持 IP 不发生变化。

5. 单击**提交**，完成 VPC 网络添加。

6. 单击操作列的**查看所有 IP 和端口**，可以查看所需要放通的 IP 和端口列表。

**说明**

如果您的服务器配置设置了访问限制（安全组），请在服务器上放通如下端口区间（因 broker 扩容、迁移后端口会发生变化，请勿仅添加当前的列表端口，避免扩容迁移后的消息读写异常。）

- VPC 路由需要放通的端口范围：9092~60000
- 公网路由需要放通的端口范围：50000~53000
- 支撑路由需要放通的端口范围：6000~12000

**IP 和端口**

如果您的服务器配置了访问限制（安全组），请在服务器上放通 6000~12000 端口区间（因 broker 扩容、迁移后端口会发生变化，请勿仅添加当前的列表端口，避免扩容迁移后消息读写异常）

当前端口（升配和底层调度时端口会变动）

9.1... 25:13010

9.1... 229:13010

9.1... 236:13010

确定

**公网域名接入**

**操作场景：**当您的消费者或者生产者处于自建机房或其他云服务时，可以通过公网访问方式对消息队列 CKafka 内的数据进行生产和消费。

**操作建议：**为了保证安全性，Kafka 提供了多种安全认证机制，主要分为 SSL 和 SASL2 大类。其中 SASL/PLAIN 是基于账号密码的认证方式，比较常用。消息队列 CKafka 支持 SASL\_PLAINTEXT 和 SASL\_SSL 认证，请您在选择公网域名接入时酌情配置接入鉴权方式。

**操作步骤：**

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击接入方式模块中的**添加路由策略**。
4. 在弹窗中，路由类型选择 **公网域名接入**，选择好接入方式。

**添加路由策略**

路由类型

公网域名接入

接入方式

SASL\_PLAINTEXT

该接入方式提供用户管理和ACL策略配置，以管理用户访问权限

公网计费模式

包年包月

公网带宽

3Mbps

CKafka默认赠送3Mbps公网带宽

提交

关闭

5. 单击**提交**，完成公网路由策略添加。

6. 单击操作列的**查看所有 IP 和端口**，可以查看所需要放通的 IP 和端口列表。

#### 说明

- 如果您的服务器配置设置了访问限制（安全组），请在服务器上放通如下端口区间（因 broker 扩容、迁移后端口会发生变化，请勿仅添加当前的列表端口，避免扩容迁移后的消息读写异常。）
  - VPC 路由需要放通的端口范围：9092~60000
  - 公网路由需要放通的端口范围：50000~53000
  - 支撑路由需要放通的端口范围：6000~12000
- CKafka 默认提供3Mbps 免费公网带宽，专业版实例可以额外升配公网带宽，具体操作参见 [升配公网带宽](#)。
- 如果选择SASL\_SSL接入方式，请 [点击下载证书](#) 后使用。

#### IP 和端口

如果您的服务器配置了访问限制（安全组），请在服务器上放通 6000~12000端口区间（因broker扩容、迁移后端口会发生变化，请勿仅添加当前的列表端口，避免扩容迁移后消息读写异常）

当前端口（升配和底层调度时端口会变动）

9.1.1.1 25:13010

9.1.1.1 229:13010

9.1.1.1 236:13010

确定

# 公网带宽管理

Last updated: 2023-04-11 09:54:24

## 操作场景

CKafka 默认内网传输，如需通过公网访问，需要单独开通一条公网路由，具体操作参见 [添加路由策略](#)，当前默认提供3Mbps 免费公网带宽。

CKafka 专业版实例支持升配公网带宽，若您有更高的带宽需求，您可以额外支付费用购买。具体价格请参见 [计费概述](#)。

本文档为您介绍在 CKafka 控制台升配调整和删除公网带宽的操作步骤。

## 操作步骤

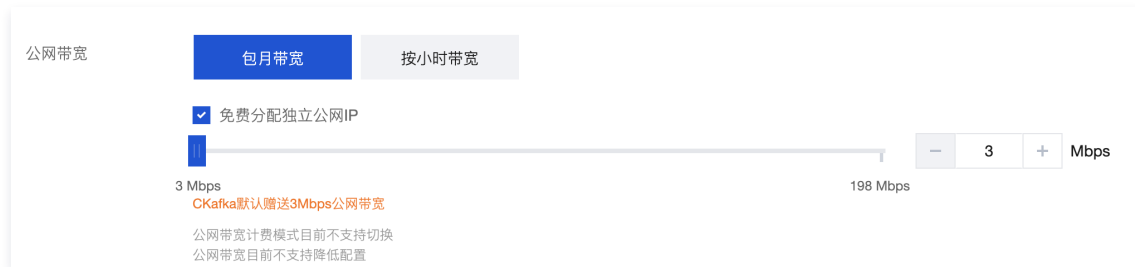
2022年01月07日起，消息队列 CKafka 对公网带宽的购买入口及购买模式进行调整，新用户可在购买实例的同时购买公网带宽，存量实例沿用按小时带宽的计费模式，可以升配。

您可以点击以下标签查看新旧实例的购买方式和相关操作方法。

### 公网带宽管理（新）

#### 购买公网带宽

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击**新建**进入购买页。
3. 在购买页，选择好实例相关信息后，根据自身业务需求选择公网带宽购买信息。
  - 计费模式：支持**包月带宽**和**按小时带宽**两种，公网带宽计费模式目前不支持切换，具体计费说明请参见 [公网带宽价格](#)。
  - 带宽大小：选择您需要购买的公网带宽大小，CKafka 默认赠送3Mbps公网带宽，您可以根据业务需求额外购买。



4. 单击**立即购买**，提交订单，完成购买。

#### 升配公网带宽

##### 说明

当前公网带宽不支持在控制台降配，请您谨慎购买。如确有降配诉求，请 [提交工单](#) 咨询。

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击**配置信息**模块公网带宽旁的**升配**。



4. 在弹窗中修改公网带宽，单击**提交**，即可完成公网带宽修改。

## 退订公网带宽

### 说明

- 当前仅按小时计费模式的公网带宽支持退订，包月计费模式的公网带宽不支持退订，可与实例一起销毁。
- 当实例下没有公网路由时，可以退订公网带宽，退订公网带宽前您需要先删除公网路由。

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击**配置信息**模块公网带宽旁的**退订**。
4. 在弹窗中单击**提交**，完成公网带宽退订。退订后，带宽将不再继续计费。



## 删除公网路由

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在接入方式模块，单击您要删除的公网带宽操作栏的**删除**，在弹窗中选择执行时间。
  - 立即执行：公网路由将即刻被删除。
  - 自定义时间：定时删除（可选择未来24小时内任意时间），公网路由将进入等待删除状态，在操作栏可修改定时删除时间。



### 说明

删除公网路由将导致负载均衡，请谨慎操作。

## 公网带宽管理（旧）

### 升配公网带宽

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击接入方式模块中的**添加路由策略**。
4. 在弹窗中，路由类型选择**公网域名接入**，选择您所需要的公网带宽大小，单击**提交**，完成购买。

**添加路由策略** ✕

路由类型 公网域名接入

接入方式 SASL\_PLAINTEXT

该接入方式提供用户管理和ACL策略配置，以管理用户访问权限

公网带宽 
3
198
-
+
Mbps

公网带宽收费价格与云服务器公网网络费用 - 按小时带宽 保持一致 [公网网络费用](#)

费用

提交
关闭

## 调整公网带宽

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击配置信息模块公网带宽旁边的编辑按钮。

**配置信息**

规格 专业版-基础型

峰值带宽 40 MB/s

磁盘容量 500GB

公网带宽 6Mbps ✎

4. 在弹窗中修改公网带宽，单击**提交**，即可完成公网带宽修改。

### 说明

公网带宽计费每小时结算，若1小时内多次更换网络带宽，以最高带宽计费。

## 删除公网路由

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在接入方式模块，单击您要删除的公网带宽操作栏的**删除**，在弹窗中选择执行时间。
  - 立即执行：公网路由将即刻被删除。
  - 自定义时间：定时删除（可选择未来24小时内任意时间），公网路由将进入等待删除状态，在操作栏可修改定时删除时间。

**接入方式** 添加路由策略

接入类型	接入方式	计划于2021-09-08 23:34:40删除 <span style="font-size: 0.8em; color: #007bff;">修改时间</span>	
公网域名接入	SASL_PLAINTEXT	ckafka-dmwq... <span style="font-size: 0.8em; color: #999;">📄</span>	等待删除 <span style="font-size: 0.8em; color: #999;">ⓘ</span>

### 说明

删除公网路由将导致负载均衡，请谨慎操作。

# 接入外部监控 Prometheus

Last updated: 2023-03-10 17:04:34

## 操作场景

腾讯云 CKafka 专业版实例默认为所有用户提供了外部监控服务的接入方式，通过提供的接入点可完成 CKafka 实例的监控，包括未同步副本、主题流入消息速率等一系列开源 Kafka 可监控的度量指标。

腾讯云 CKafka 专业版实例目前提供 Prometheus 抓取监控数据的 broker 节点指标信息，包括有 CPU、内存使用情况、系统负载等基本监控度量指标，以及 broker JMX 暴露出的度量指标。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择实例列表，单击目标实例的“ID”，进入实例基本信息页面。
3. 在使用 Prometheus 监控模块单击右上角的获取监控目标，选择 VPC 和子网。

### 获取监控目标

网络  0/1  0.0

如果现有的网络不合适，您可以去控制台[新建私有网络](#)或[新建子网](#)

4. 单击提交，获取一组监控目标。

### 用Prometheus监控实例 获取监控目标

jmx exporter	node exporter	操作
10.0.0. :60001 <input type="button" value="删除"/>	10.0.0. :60002 <input type="button" value="删除"/>	
10.0.0. :60003 <input type="button" value="删除"/>	10.0.0. :60004 <input type="button" value="删除"/>	删除
10.0.0. :60005 <input type="button" value="删除"/>	10.0.0. :60006 <input type="button" value="删除"/>	

5. 下载 [Prometheus](#)，并配置监控抓取地址。
  - 5.1 进入 Prometheus 程序包所在目录，执行如下命令，解压 Prometheus 程序包。

```
tar -vxf prometheus-2.30.3.linux-amd64.tar.gz
```

- 5.2 修改配置文件 prometheus.yml，增加 jmx\_exporter 与 node\_exporter 抓取任务。

```
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: "prometheus"
    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.
    static_configs:
      - targets: ["localhost:9090"]

  - job_name: "broker-jmx-exporter"
    scrape_interval: 5s
    metrics_path: '/metrics'
    static_configs:
      - targets: ['10.x.x.0:60001','10.x.x.0:60003','10.x.x.0:60005']
      labels:
        application: 'broker-jmx'
  - job_name: "broker-node-exporter"
    scrape_interval: 10s
```

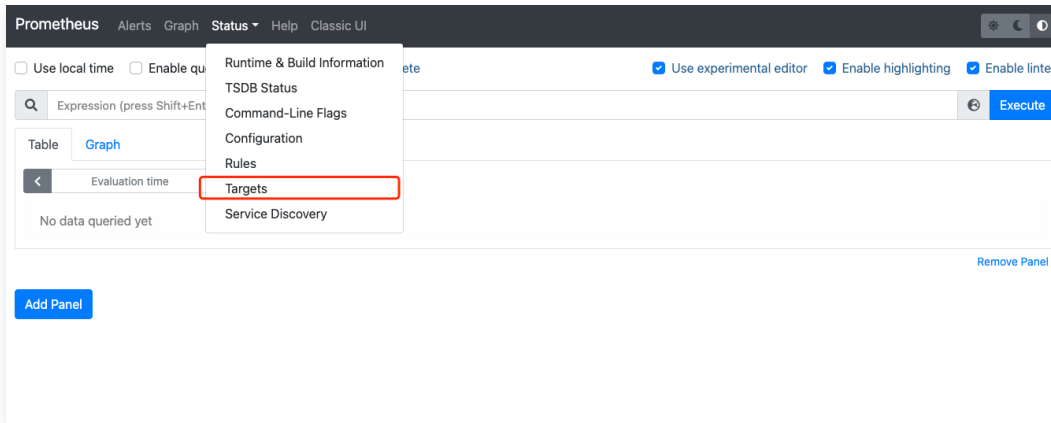
```
metrics_path: '/metrics'
static_configs:
- targets: ['10.x.x.0:60002','10.x.x.0:60004','10.x.x.0:60006']
  labels:
    application: 'broker-node'
```

其中 `broker-jmx-exporter` 是 Prometheus 抓取 broker 的 jmx 指标配置的标签项，Targets 其中包含映射的端口信息，而 `broker-node-exporter` 为抓取 broker 所在节点的基本指标配置的标签项，`scrape_interval` 为抓取监控度量数据的频率。

### 5.3 启动 Prometheus。

```
./prometheus --config.file=prometheus.yml --web.enable-lifecycle
```

### 5.4 打开 Prometheus 提供的 UI 界面查看接入的 Targets 状态是否正常，如在浏览器输入 `http://localhost:9090`。



### 5.5 检查 Targets 状态都是 UP。

The screenshot shows the Prometheus web interface displaying the 'Targets' page. The table lists the status of various targets, all of which are 'UP'.

Endpoint	State	Labels	Last Scrape	Scrape Duration
<b>broker-jmx-exporter (2/2 up)</b> <a href="#">show less</a>				
<code>http://10.0.0.1:60003/metrics</code>	UP	<code>application="broker-jmx"</code> <code>instance="10.0.1.176:60003"</code> <code>job="broker-jmx-exporter"</code>	7.171s ago	0.819ms
<code>http://10.0.0.1:60001/metrics</code>	UP	<code>application="broker-jmx"</code> <code>instance="10.0.1.176:60001"</code> <code>job="broker-jmx-exporter"</code>	5.206s ago	1.464ms
<b>broker-node-exporter (1/1 up)</b> <a href="#">show less</a>				
<code>http://10.0.0.1:60002/metrics</code>	UP	<code>application="broker-node"</code> <code>instance="10.0.1.176:60002"</code> <code>job="broker-node-exporter"</code>	4.241s ago	8.333ms

如果 Targets 状态为 DOWN 则需要检查网络访问是否可达，或根据状态栏最后的 Error 选项查看原因。

### 6. 查询监控指标数据。

单击 Graph 选项输入查询的指标名称即可看到相应的监控数据，例如输入 `node_memory_MemAvailable_bytes`，单击 **execute**。

The screenshot shows the Prometheus web interface. At the top, there is a navigation bar with 'Prometheus', 'Alerts', 'Graph', 'Status', and 'Help'. Below this, there are several checkboxes: 'Use local time' (unchecked), 'Enable query history' (unchecked), 'Enable autocomplete' (checked), 'Use experimental editor' (checked), 'Enable highlighting' (checked), and 'Enable linting' (checked). A search bar contains the query 'node\_memory\_MemAvailable\_bytes' and an 'Execute' button. Below the search bar, there are tabs for 'Table' and 'Graph'. The 'Table' view is active, showing a table with one row of data. The table has columns for 'Evaluation time' and 'Result series'. The data row shows the query 'node\_memory\_MemAvailable\_bytes(application="broker-node", instance="...00", job="broker-node-exporter")' and the value '11712106496'. There are 'Add Panel' and 'Remove Panel' buttons at the bottom of the table view.

Evaluation time	Result series
	node_memory_MemAvailable_bytes(application="broker-node", instance="...00", job="broker-node-exporter") 11712106496

# 迁移可用区

Last updated: 2023-05-29 11:11:24

## 操作场景

您可以将消息队列 CKafka 专业版实例迁移至同一地域内的其它可用区。迁移可用区后，实例的所有属性、配置和连接地址都不会改变。迁移所需时间跟实例的数据量有关。

例如在如下场景中，您可以选择迁移可用区：

- 假设您正在尝试修改实例的实例类型，但我们无法在当前可用区中启动新实例类型的实例。在这种情况下，您可以将实例迁移到能够启动该实例类型的可用区。
- 当前可用区已无资源进行扩容的情况下，您也可以将实例迁移至同地域内其他资源充足的可用区，以满足业务需要。

## 前提条件

- 实例状态为运行中。
- 实例所在的地域需要有多个可用区，才支持迁移可用区功能。

## 费用说明

本功能免费。即使将实例从单可用区迁移至多个可用区，也不收取费用。

## 功能说明

- 当原实例是单可用区部署时，可以切换可用区，也可以升级为多可用区部署。关于多可用区部署详情请参见 [跨可用区部署](#)。
- 当原实例是多可用区部署时，不支持切换可用区，也不支持切换回单可用区部署。

## 迁移类型

迁移类型	场景
从一个可用区迁移至另一个可用区	实例所在可用区出现满负载或者其它影响实例性能的情况。
从一个可用区迁移至多个可用区	提高实例的容灾能力，实现跨机房容灾。主备实例分别位于不同的可用区。相对于单可用区实例，多可用区实例可以承受更高级别的灾难。例如，单可用区实例可以承受服务器和机架级别的故障，而多可用区实例可以承受机房级别的故障。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击实例列表，单击目标实例的“ID/名称”，进入基本信息页。
3. 在基本信息模块，单击可用区右边的编辑按钮，选择您要选择的切换的可用区。



4. 单击确认，预计等待5-10分钟完成变配，在实例列表的状态栏可以查看变配进度。



# 设置系统维护时间

Last updated: 2023-04-11 09:54:24

## 操作场景

维护时间对于消息队列 CKafka 而言非常重要，为保证您的 CKafka 实例的稳定性，后台系统会不定期在维护时间内对实例进行维护操作。建议您对业务实例设置自己可接受的维护时间，一般设置在业务低峰期，将对业务的影响降到最低。

### 说明

- 维护时间默认值是晚上23:00，持续时间1小时。CKafka 专业版实例支持修改系统维护时间，标准版实例仅作参考，不支持修改。
- 消息队列 CKafka 在进行维护前，会向腾讯云账户内设置的联系人发送短信和邮件，请注意查收。

## 设置维护时间

- 登录 [CKafka 控制台](#)。
- 在左侧导航栏单击实例列表，单击目标实例的“ID/名称”，进入基本信息页。
- 在基本信息模块，单击系统维护时间旁的编辑按钮。
- 在弹出的对话框，选择您所需的“维护周期”和“维护时间”。

### 修改维护周期和时间

维护周期  星期一  星期二  星期三  星期四  星期五  
 星期六  星期日

维护时间 开始时间   UTC+8:00

持续时间  小时

- 单击确定，完成设置。

## 配置消息大小

Last updated: 2022-10-28 09:32:25

### 操作场景

CKafka 支持实例级别设置默认消息大小，作为新建 Topic 时的默认初始值，可以后期单独针对 Topic 进行修改。

### 设置默认消息大小

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面，单击消息配置模块中的**修改配置**。
4. 设置消息保留时长和最大消息大小后，单击提交。

#### 消息配置

消息保留时长  天 消息保留范围在1分钟到90天

最大消息大小  MB 范围在1 KB到12 MB  
新建Topic的默认初始值，可以后期单独针对Topic进行修改

### 配置单 Topic 消息大小

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **topic 管理** 标签页，在操作栏单击**编辑**。
4. 在编辑 Topic 页面，点开高级配置，在 **max.message.bytes** 处可修改 Topic 最大消息大小。

max.message.bytes  MB 客户端发送数据时，会将发往同一个分区的数据聚合起来，统一发送，服务端会比较每一批次的消息大小，范围1 KB到12 MB

# Topic 管理

## 实例 Topic

### 创建 Topic

Last updated: 2023-04-11 09:54:25

#### 操作场景

Topic（主题）是某一类分类的名字，消息在 Topic 中可以被存储和发布。CKafka 对外使用 Topic 的概念，生产者往 Topic 中写消息，消费者从 Topic 中读消息。为了做到水平扩展，一个 Topic 实际是由多个 **Partition（分区）** 组成，遇到瓶颈时，可以通过增加 Partition 的数量进行横向扩容。

本文介绍在 CKafka 控制台创建一个 Topic 的操作步骤。

#### 前提条件

[已创建实例](#)。

#### 操作步骤

##### 步骤1: 创建 Topic

1. 登录 [CKafka 控制台](#)。
2. 在实例列表页，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击页面顶部的 **Topic 管理**，单击**新建**。
4. 在编辑 Topic 窗口中，选择分区数和副本数等信息。

#### 新建Topic

名称

备注

分区数  1 3000    个  
单个Topic支持最大分区数：3000  
[分区数配置建议](#)

副本数  1  2个副本  3  
选择n个副本时，最多允许有(n-1)台broker宕机  
实例支持最大分区\*副本数：900，当前额度已用292个，实例还可最多创建304个2副本分区  
如需更多分区，可操作实例升配，具体规则见[文档](#)

标签 [+ 添加](#)  
标签用于从不同维度对资源分类管理。如现有标签不符合您的要求，请前往控制台[管理标签](#)

预设ACL策略

[展示高级配置](#)

- 名称：Topic 名称，输入后无法更改，名称只能包含字母、数字、下划线、“-”和“.”。
- 分区数：一个物理上分区的概念，一个 Topic 可以包含一个或者多个 partition，CKafka 以 partition 作为分配单位。分区数配置参考文档 [参数配置说明](#)
- 副本数：Partition 的副本个数，用于保障 Partition 的高可用。为保障数据可靠性，默认开启2副本。副本数也算分区个数，例如客户创建了1个 Topic、6个分区、2个副本，那么分区额度一共用了  $1 \times 6 \times 2 = 12$  个。

**注意**

设置为单副本会导致可用性无法保证，请谨慎操作。

- 标签：设置资源标签，关于标签的详细介绍请参见 [标签管理](#)。
- 预设 ACL 策略：勾选提前设置好的 ACL 策略，关于 ACL 策略详情请参见 [配置 ACL 策略](#)。

5. 单击**提交**，完成 Topic 创建。

**步骤2：配置 Topic 高级参数**

1. 在实例列表页，单击目标实例的“ID/名称”，进入实例详情页。
2. 在实例详情页，单击 **topic 管理** 标签页。
3. 单击操作列的**编辑** > **展示高级配置**，设置如下参数：

cleanup.policy	<input type="text" value="delete"/>	支持日志按保存时间删除，或者日志按key压缩（kafka connect时需要使用compact模式）
min.insync.replicas	<input type="text" value="1"/>	当producer设置request.required.acks为-1时，min.insync.replicas指定replicas的最小数目
unclean.leader.election.enable	<input checked="" type="checkbox"/>	
segment.ms	<input type="text" value=""/> <input type="text" value="ms"/>	Segment分片滚动的时长，范围1 到90 天
retention.ms	<input type="text" value="2"/> <input type="text" value="天"/>	topic维度的消息保留时间，范围1 分钟到90 天
retention.bytes	<input type="text" value=""/> <input type="text" value="B"/>	分区维度的消息保留大小，范围1 到1024 GB 分区数 * retention.bytes = 当前topic的消息保留大小，对于一个 topic，如果同时设置了消息保留时间和消息保留大小，实际保留消息时会以先达到的阈值为准。
max.message.bytes	<input type="text" value=""/> <input type="text" value="B"/>	客户端发送数据时，会将发往同一个分区的数据聚合起来，统一发送，服务端会比较每一批次的消息大小，范围1 KB到12 MB

参数说明如下：

参数名	默认值	参数范围	说明
cleanup.policy	delete	delete/compact	支持日志按保存时间删除，或者日志按 key 压缩（Kafka Connect 时需要使用 compact 模式）。
min.insync.replicas	1	-	当 producer 设置 request.required.acks 为1时，min.insync.replicas 指定 replicas 的最小数目。
unclean.leader.election.enable	true	true/false	指定是否能够设置不在 ISR 中 replicas 作为 leader。
segment.ms	-	5mins - 90days	Segment 分片滚动的时长，单位为 ms，最小值为86400000ms。
retention.ms	默认为实例的消息	60000ms -	Topic 维度的消息保留时间。

	保留时间	90days	
retention.bytes	默认为实例的消息保留大小	1GB - 1024GB	分区数 * retention.bytes = 当前topic的消息保留大小，对于一个Topic，如果同时设置了消息保留时间和消息保留大小，实际保留消息时会以先达到的阈值为准。
max.message.bytes	-	1KB - 12MB	Topic 维度的最大消息大小。不填写则默认实例维度消息大小为1MB。

# 查看 Topic

Last updated: 2023-04-11 09:54:25

## 操作场景

本文介绍您在 CKafka 控制台创建 Topic 后，查看 Topic 详情和生产端连接关系的操作步骤。

## 操作步骤

### 步骤1: 查看 Topic 详情

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **topic 管理** 标签页，查看 Topic 信息，进入 Topic 列表页。
4. 在 Topic 列表页，单击 Topic 名称左侧右三角符号，查看 Topic 详情。

ID/名称	监控	分区数(个)	副本数(个)	标签	备注	创建时间	消息保留时间	状态	操作
topic- □		8	2			2021-12-28 11:13:23	1天	正常	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">更多</a>

分区名称	leader	副本	ISR	起始offset	末端offset	消息数	未同步副本
partition-0	102474	102474,102475	102474,102475	288	288	0	-
partition-1	102475	102475,102476	102476,102475	1	1	0	-
partition-2	102476	102476,102473	102476,102473	0	0	0	-

项目	说明
分区名称	partition 的名称
leader	leader 处理 partition 的所有读写请求，follower 会被定期地去复制 leader 上的数据
副本	副本列表
ISR	已同步消息的副本
起始 offset	消息最后消费的位置
末端 offset	消息最后写入的位置
消息数	存储的消息数量
未同步副本	未同步的副本数量，支持筛选存在未同步副本的 partition

### 步骤2: 查看生产端连接关系

#### ① 说明

当前仅2.4版本及以上专业版实例支持查看生产端连接关系。

1. 在实例列表页，单击目标实例的“ID/名称”，进入实例详情页。
2. 在实例详情页，单击 **topic 管理** 标签页，在操作栏单击**生产端连接关系**，查看与 Topic 连接的生产者列表信息。

ckafka-pkwxedpq

生产端连接关系

topic管理 Consumer Group 监控 事件中心

新建(24/450)

ID/名称	监控	分区数(个)	副本数(个)
topic-1		3	2

分区名称	leader	副本
partition-0	117953	117953,117955

客户端IP 连接时间

当前没有生产者连接

共 0 条 10 条 / 页 1 / 1 页

# 删除 Topic

Last updated: 2023-04-11 09:54:25

## 操作场景

当您不再需要某 Topic 时，您可以在 CKafka 控制台手动删除该 Topic。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **topic 管理** 标签页，在操作栏单击**删除**。
4. 在弹出窗口单击**删除**，Topic 将被删除。

### ⚠ 注意

- 删除 Topic 的同时，存储在此 Topic 中的消息也将被删除，请谨慎操作。
- Topic 删除是异步操作，配置删除成功后，ZooKeeper 配置将会在1分钟后生效。若此期间创建同名 Topic，系统会提示错误码 [4000]10011，届时请您稍后重试。

# 发送消息

Last updated: 2023-09-06 16:09:51

## 操作场景

您在 CKafka 控制台创建 Topic 后，可以进行发送消息测试，并在消息查询中查看刚刚发送消息的记录。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **topic 管理** 标签页，在操作栏单击**发送消息**。

### 发送消息

消息内容  0 / 1024

仅用于调试，消息内容不可超过1024个字符。

消息key

发送到指定分区

- 消息内容：填写发送消息内容，必填。
  - 消息Key：填写发送 Key，选填。
  - 发送到指定分区：支持将消息发送到指定分区，默认关闭。
4. 单击**确认**，发送消息。在消息发送成功弹窗中单击**消息查询**可以查看刚刚发送的消息。

# 设置 Topic 限流规则

Last updated: 2023-03-28 10:49:14

## 操作场景

您可以针对 Topic 设置限流规则，避免单个 Topic 流量过大而影响其他 Topic。

### 说明

只有 broker 版本为1.1.1、2.4.1和2.8.1才支持 topic 设置限流规则。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择实例列表，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **topic 管理** 标签页。
4. 单击操作列的**更多** > **限流**，设置限流阈值。

- topic 最大生产流量：不含副本流量，取值范围为1MB/s到该实例购买的最大带宽/该 Topic 副本数。
- topic 最大消费流量：取值范围为1MB/s到该实例购买的最大带宽。

### 说明

- 底层针对 broker 进行限流，实际限流值（等于 broker 数量的整数倍）可能会与设置的限流值略有区别。
- 关于软限流机制说明请参见 [限流说明](#)。

# 弹性 Topic

## Topic 管理

Last updated: 2023-07-13 17:24:31

### 操作场景

Topic（主题）是某一类名字，消息在 Topic 中可以被存储和发布，生产者往 Topic 中写消息，消费者从 Topic 中读消息。为了做到水平扩展，一个 Topic 实际是由多个 Partition（分区）组成，遇到瓶颈时，可以通过增加 Partition 的数量进行横向扩容。Topic 是连接器订阅和发布的最小单位，用户可以用 Topic 来表示一类或者一种流数据。

CKafka 支持弹性 Topic 类型，您可以在控制台直接创建弹性 Topic 并且不需要先购买 CKafka 实例，使用弹性 Topic，您同样可以使用 SDK 完成消息收发，同时也可以将弹性 Topic 作为数据任务的数据源或者数据目标。

#### 说明：

当前弹性 Topic 暂未开启计费，后续开启收费将会提前通知您。

### 使用限制

- 单弹性 Topic 支持的分区数量上限为500个，弹性 Topic 带宽无默认大小，带宽建议使用不要超过500M。
- 弹性 Topic 支持最大消息大小为 12MB

### 操作步骤

#### 创建 Topic

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击弹性 Topic，选择好地域后，单击新建 Topic。
3. 填写 Topic 信息后，单击确定，完成 Topic 创建。
  - Topic 名称：Topic 名称由“appid-[用户输入的名称]”构成，用户输入的名称只能包含字母、数字、下划线、“-”、“.”。
  - 备注：选填，Topic 备注信息。
  - 分区数：一个物理上分区的概念，一个 Topic 可以包含一个或者多个 Partition，CKafka 以 Partition 作为分配单位。单个 Topic 支持最大分区数：500。
  - 消息保留时间：范围1分钟到90天。在磁盘容量不足（即磁盘水位达到90%）时，将会提前删除旧的消息，以确保服务可用性。

The screenshot shows the 'Create Topic' dialog box in the CKafka console. It contains the following fields and controls:

- Topic名称**: Input field with value '1300957330-' and a placeholder '请输入Topic名称'.
- 备注**: Input field with placeholder '选填，请输入备注信息'.
- 分区数**: A slider control ranging from 1 to 500. The current value is 1. Below the slider, it says '单个Topic支持最大分区数：500'.
- 消息保留时间**: A dropdown menu set to '1' with a unit of '天' (days). Below it, the range is specified as '范围1分钟到90天'.
- Buttons: '确定' (Confirm) and '取消' (Cancel).

#### 查询 Topic

在 [弹性 Topic](#) 列表右上角的搜索框中，通过 Topic 名称进行搜索查询，连接器将会模糊匹配并呈现搜索结果。

#### 编辑 Topic

1. 在 [弹性 Topic](#) 列表中，找到需要编辑的 Topic，单击操作栏中的编辑。
2. 在弹出的对话框中可以对 Topic 的备注，分区数和消息保留时间进行修改。
3. 单击确认即完成对 Topic 的编辑。

#### 发送消息

CKafka 控制台支持手动发送消息，在控制台进行相应的操作即可实现消息发送给指定的 Topic。

1. 在 [弹性 Topic](#) 列表中，找到需要发送消息的 Topic，单击操作列中的 [发送消息](#)。
2. 在弹出的对话框中输入消息内容和消息 key，同时您也可以将消息发送到指定分区。

### 发送消息

消息内容

消息key

发送到指定分区

分区ID

[确认](#) [关闭](#)

3. 单击 [确认](#)，完成消息的发送。消息发送后即可被 Topic 下的任意订阅者消费。

## 新增订阅

支持手动创建订阅，在控制台进行相应的操作后即可完成订阅的创建。

1. 在 [弹性 Topic](#) 列表中，找到需要创建订阅的 Topic，单击 Topic 的 ID 进入 Topic 基本信息页面。
2. 在页面上方选择 [订阅关系](#) 页签，单击 [新建订阅关系](#)，在弹出的对话框中输入消费者名称。

### 新建订阅关系

消费者名称

[提交](#) [关闭](#)

3. 单击 [提交](#) 完成创建，即可在列表中看到刚刚创建的订阅。

## 删除 Topic

### 注意

删除了 Topic 之后也会清除该 Topic 下积累的未消费消息，请谨慎执行。

1. 在 [弹性 Topic](#) 列表中，找到需要删除的 Topic，单击操作列中的 [删除](#)。
2. 在弹出的提示框中，单击 [删除](#)，完成删除。

### 确认删除当前所选Topic?

删除后，Topic中的消息也将被删除

[删除](#) [取消](#)

# 订阅管理

Last updated: 2023-04-11 09:54:25

## 操作场景

订阅代表一个具体的消费者以及其对某个 Topic 的订阅关系。当一个消费者订阅了某个 Topic 之后，则该 Topic 下的消息均可以被其消费。CKafka 连接器的订阅服务提供了服务端保存用户消费点位的功能，本文介绍在 CKafka 控制台新增订阅关系和重置消费点位的操作方法。

## 前提条件

已 [创建 Topic](#)。

## 操作步骤

### 新增订阅

请参见 [Topic 管理-新增订阅](#)。

### 设置 offset

在离线数据处理等场景下，有时需要对 offset 进行重置，用于消费前一时间段的消息。

1. 在 [Topic 列表](#) 中，找到需要创建订阅的 Topic，单击 Topic 的 ID 进入 Topic 基本信息页面。
2. 在页面上方选择 [订阅关系](#) 页签，单击目标订阅操作列的 [offset 设置](#)。
3. 在 offset 设置窗口，选择以 Topic 或者 Partition 为维度进行设置，单击 [下一步](#)。
  - 以 Topic 为维度设置：适用于批量重置消费位点，按业务设置消费位点等场景。
  - 以 Partition 为维度设置：适用于在了解各个 Partition 消费情况后，针对分区设置消费位点。
4. 选择需要重置 offset 的 Topic 信息（不选则默认全部 Topic 的 offset 均重置）或者 Partition 信息，单击 [下一步](#)。
5. 对 offset 进行指定。



### 注意

- offset 设置范围要在最小 offset 和最大 offset 之间。在配置时，如果小于最小 offset 会从最小 offset 进行消费，如果大于最大 offset 会从最大 offset 进行消费。
- 重置消费分组时，需保证没有消费者在消费分组内才能进行重置，否则不能进行重置。

## 查看消费者详情

- 在订阅关系列表页，单击操作列的 [查看消费者详情](#)，可以查看该消费组中的消费者信息，具体消费者和订阅 Topic 的对应关系。
- 在订阅关系列表页，单击消费者名称列左侧的小三角，可以展示该消费者组订阅的主题信息，包含主题名称、分区数目、提交的 offset 位置，最大的 offset 位置以及未消费消息条数等。

单击操作列的 [查看详情](#) 可以看到分区级别的 offset 消费情况。

新建订阅关系					
消费组名称	状态	协议类型	均衡算法	操作	
topic-1-test	Empty	consumer	-	<a href="#">offset设置</a> <a href="#">查看消费者详情</a> <a href="#">删除</a>	

请输入主题名搜索					
分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作	
partition-0	0	0	0	<a href="#">查看详情</a>	

**注意**

由于 offset 信息是在消费端维护的，因此 offset 的位置和消费者提交 offset 的方式有关，是异步展示的，并不一定代表实时的消费情况。

**删除订阅**

在订阅关系列表中，单击目标消费者操作栏的删除，二次确认后可直接删除该消费者。

**注意：**

删除消费者组后，其中的消费者重新建立消费连接时，会重置 offset，即从头开始消费。

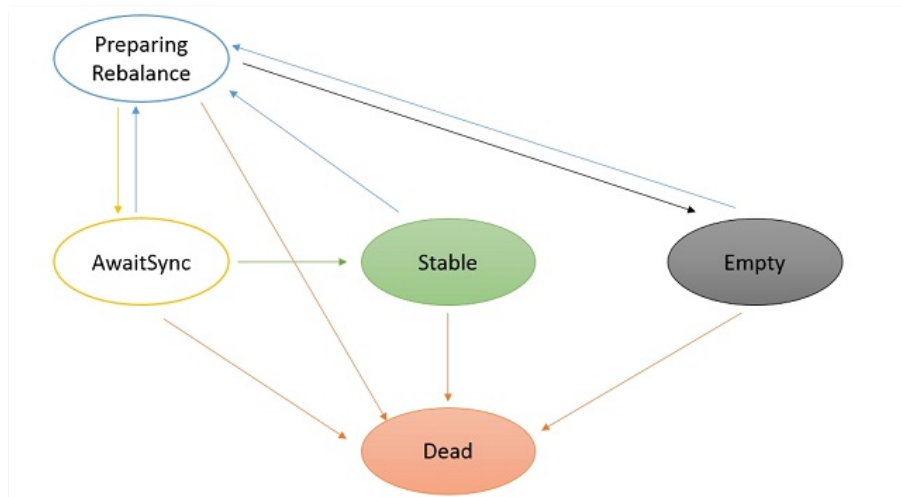
# Consumer Group

## 概述

Last updated: 2021-01-28 10:08:44

### Consumer Group 状态说明

消费者组列表页中 Consumer Group 的状态主要有 Dead、Empty、PreparingRebalance、AwaitingSync、Stable 几种，其中最常见的是 Empty、Stable 和 Dead 三种状态。Consumer Group 中的状态机转换如下图所示：



- Dead: 消费者组内无成员并且 Metadata 已经被移除。
- Empty: 消费分组内当前没有任何成员。如果组内所有 offset 都已过期，则会变为 Dead 状态。一般新创建的 Group 默认为 Empty 状态。开源 Kafka 0.10.x 版本规定，当消费分组内没有任何成员且状态持续超过7天，此消费分组将会被自动删除。
- Stable: 消费分组中各个消费者已经加入，处于稳定状态。

### Rebalance 状态详解

#### Rebalance 发生原因

根据 Consumer Group 的状态机可知，当 Consumer Group 为 Empty、AwaitingSync 或 Stable 状态时，Group 可能会进行 Rebalance。以下情况可能会发生 Rebalance：

- 一个消费者订阅了 Topic。
- 消费者被关闭。
- 某个 Consumer 被 Group Coordinator (协调器) 认为是 Dead 状态时。  
如果某个 Consumer 在 `session.timeout.ms` 时间内没有给 Group Coordinator 发心跳，则该 Consumer 将被认为是 Dead 状态，并且发起 Rebalance。详情请参见 [CKafka 常用参数配置指南](#)。
- 分区数增加。
- 订阅了不存在的 Topic。  
如果您订阅了一个还未创建的 Topic，那么当这个 Topic 创建后会发生 Rebalance；同理，如果一个已经被订阅的 Topic 被删除，也会发生 Rebalance。
- 应用崩溃。

#### Rebalance 过程分析

以0.10版本Kafka的机制为例，Rebalance过程分析如下：

1. 任何一个 Consumer 想要加入到一个 Consumer Group 中时，会发送一个 JoinGroup 的请求给 Group Coordinator。第一个加入 Group 的 Consumer 会变成 Group Leader。
2. Leader 会从 Group Coordinator 处收到这个 Group 中所有 Consumer 列表，并且负责给 Group 中的 Consumer 分配 partition。分区的分配可以通过 PartitionAssignor 接口来实现。
3. 分配完成后，Leader 会把分配结果发给 Group Coordinator，Coordinator 会把结果发送给所有的 Consumer。  
因此，每个 Consumer 只能查看到自己被分配的 partition，Leader 是唯一能够拿到 Consumer Group 中的 Consumer 以及其分区情况的节点的 Consumer。

上述过程会在每次 Rebalance 发生时执行一次。

# 创建 Consumer Group

Last updated: 2023-04-12 09:38:54

## 操作场景

消息队列 CKafka 支持在控制台直接创建 Consumer Group，本文介绍在控制台创建 Consumer Group 的具体步骤。

### 说明

单个实例建议不超过 200 个消费分组，超出会有一定限制。

## 操作步骤

### 创建 Consumer Group

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，选择 **Consumer Group** 标签页，单击**新建消费组**。



4. 在弹窗中填写消费组名称，消费分组名称不能包含英文字符下的 `"`，勾选所要订阅的 Topic。

### 说明

支持同时勾选多个 Topic。

5. 单击**提交**，在 Consumer Group 列表可以看到刚刚创建的消费者组。

### 关闭自动创建 Consumer Group

CKafka 默认允许自动创建 Consumer Group，您可以在 CKafka 控制台关闭允许自动创建 Consumer Group 开关，关闭后只能消费在控制台上已有的消费组，无法正常新建数据同步任务。



### 说明

仅支持专业版和2.4.1及以上版本。

# 查询 Consumer Group

Last updated: 2023-04-11 09:54:25

## 操作场景

该任务指导您在 CKafka 控制台查看实例下的消费者组信息。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **Consumer Group** 标签页，查看到当前 CKafka 实例的消费组信息。



- 在 Consumer Group 列表页，单击操作列的**查看消费者详情**，可以查看该消费组中的消费者信息，具体消费者和订阅 topic 的对应关系。
- 在 Consumer Group 列表页，单击消费者名称列左侧的小三角，可以展示出该消费者组订阅的主题信息，包含主题名称、分区数目、提交的 offset 位置，最大的 offset 位置以及未消费消息条数等。单击操作列的**查看详情**可以看到分区级别的 offset 消费情况。

### 说明

由于 offset 信息是在消费端维护的，因此 offset 的位置和消费者提交 offset 的方式有关，是异步展示的，并不一定代表实时的消费情况。

# 删除 Consumer Group

Last updated: 2023-04-11 09:54:26

## 操作场景

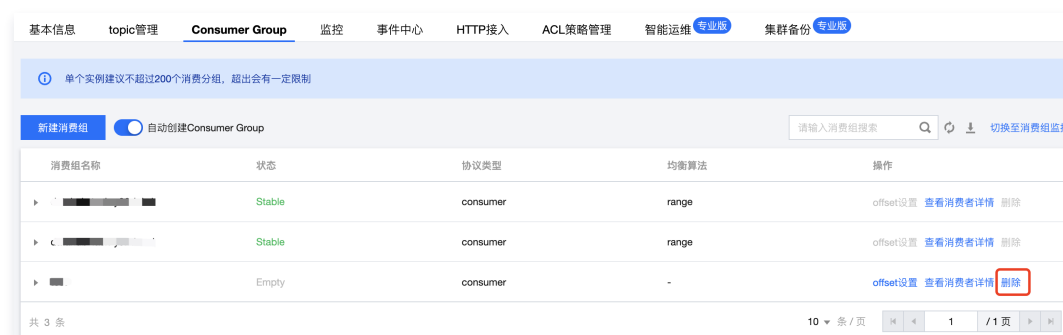
部分场景下，Consumer Group 会很长一段时间不消费后重新消费，可将消费者组删除，其中的消费者重新建立连接时，会重置 Offset，从头开始消费。

## 删除 Consumer Group

### 说明


Broker 版本不低于1.1.1，且 Consumer Group 的状态为 Empty 时，消费组才能被删除。

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击目标 Consumer Group 操作栏的**删除**，可直接删除 Consumer Group。



## 删除关联 Topic 的订阅关系

当 Consumer group 状态为 Empty 时，支持删除关联的某个 Topic 的订阅关系。操作方法如下：

在 Consumer Group 列表页面，点击要删除的订阅关系旁的  图标，确认后即可删除所关联 Topic 的订阅关系。



# 设置 Offset

Last updated: 2021-12-17 16:48:51

## 操作场景

在离线数据处理等场景下，有时需要对 offset 进行重置，用于消费前一时间段的消息。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，单击 **Consumer Group** 标签页，单击操作列的 **offset 设置**。
4. 在 offset 设置窗口，选择以 Topic 或者 Partition 为维度进行设置，单击**下一步**。
5. 选择需要重置 offset 的 Topic 信息（不选则默认全部 Topic 的 offset 均重置）或者 Partition 信息，单击**下一步**。
6. 对 offset 进行指定。

### offset设置

选择方式 > 选择对象 > 3 Offset设置

offset设置  将offset移动到指定位置

offset的调整范围介于最小offset和最大offset之间  
如输入位置小于/大于offset范围则会重置到最小/最大offset位置

将offset向前或向后移动若干条

从最新 / 最开始位置开始消费

按时间点进行消费位置重置

### ⚠ 注意

- offset 设置范围要在最小 offset 和最大 offset 之间。在配置时，如果小于最小 offset 会从最小 offset 进行消费，如果大于最大 offset 会从最大 offset 进行消费。
- 重置消费分组时，需保证没有消费者在消费分组内才能进行重置，否则不能进行重置。

# 自动重置 offset

Last updated: 2023-06-29 09:38:31

本文主要介绍 auto.offset.reset 参数的相关概念及使用方式。

## 什么是 auto.offset.reset?

auto.offset.reset 参数定义了当无法获取消费分区的位移时从何处开始消费。例如：当 Broker 端没有 offset（如第一次消费或 offset 超过7天过期）时如何初始化 offset，当收到 OFFSET\_OUT\_OF\_RANGE 错误时如何重置 Offset。

auto.offset.reset 参数设置有如下选项：

- earliest：表示自动重置到 partition 的最小 offset。
- latest：默认为 latest，表示自动重置到 partition 的最大 offset。
- none：不自动进行 offset 重置，抛出 OffsetOutOfRangeException 异常。

## 什么时候会出现 OFFSET\_OUT\_OF\_RANGE?

该错误表示客户端提交的 offset 不在服务端允许的 offset 范围之内。例如：topicA 的分区1的 LogStartOffset 为100，LogEndOffset 为300，此时如果客户端提交的 offset 小于100或者大于300，服务端就会返回该错误，此时就会进行 offset 重置。

以下情况可能会导致客户端触发该错误：

- 客户端设置了 offset，然后一段时间内没有消费，但 Topic 设置了消息保留时间，当过了保留时间后，offset 在服务端已经被删除了，即发生了日志滚动，此时客户端再提交删除了的 offset，则会发生该错误。
- 因为 SDK Bug、网络丢包等问题，导致客户端提交了异常的 offset，则会触发该错误。
- 服务端有未同步副本，此时发生了 leader 切换，触发了 follower 副本的截断，此时如果客户端提交的 offset 在截断的范围之内，则会触发该错误。

## auto.offset.reset=none 使用说明

### 使用背景

不希望发生 offset 自动重置的情况，因为业务不允许发生大规模的重复消费。

#### 注意

此时消费组在第一次消费的时候就会找不到 offset 而报错，这时就需要在 catch 里手动设置 offset。

### 使用说明

auto.offset.reset 设置为 None 以后，可以避免 offset 自动重置的问题，但是当增加分区的时候，因为关闭了自动重置机制，客户端不知道新的分区要从哪里开始消费，则会产生异常，此时需要人工去设置消费分组 offset 并消费。

### 使用方式

消费者在消费时，当 consumer 设置 auto.offset.reset=none，捕获到 NoOffsetForPartitionException 异常，在 catch 里自己设置 offset。您可以根据自身业务情况选择以下方式中的其中一种。

- 指定 offset，这里需要自己维护 offset，方便重试。
- 指定从头开始消费。
- 指定 offset 为最近可用的 offset。
- 根据时间戳获取 offset，设置 offset。

示例代码如下：

```
package com.tencent.tcb.operation.ckafka.plain;

import com.google.common.collect.Lists;
import com.tencent.tcb.operation.ckafka.JavaKafkaConfigurer;
import java.time.Instant;
import java.time.temporal.ChronoUnit;
import java.util.ArrayList;
import java.util.Collection;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
```

```
import java.util.Map.Entry;
import java.util.Properties;
import org.apache.kafka.clients.CommonClientConfigs;
import org.apache.kafka.clients.consumer.ConsumerConfig;
import org.apache.kafka.clients.consumer.ConsumerRecord;
import org.apache.kafka.clients.consumer.ConsumerRecords;
import org.apache.kafka.clients.consumer.KafkaConsumer;
import org.apache.kafka.clients.consumer.NoOffsetForPartitionException;
import org.apache.kafka.clients.consumer.OffsetAndTimestamp;
import org.apache.kafka.clients.producer.ProducerConfig;
import org.apache.kafka.common.PartitionInfo;
import org.apache.kafka.common.TopicPartition;
import org.apache.kafka.common.config.SaslConfigs;

public class KafkaPlainConsumerDemo {

    public static void main(String args[]) {
        //设置JAAS配置文件的路径。
        JavaKafkaConfigurer.configureSaslPlain();

        //加载kafka.properties。
        Properties kafkaProperties = JavaKafkaConfigurer.getKafkaProperties();

        Properties props = new Properties();
        //设置接入点，请通过控制台获取对应Topic的接入点。
        props.put(ProducerConfig.BOOTSTRAP_SERVERS_CONFIG, kafkaProperties.getProperty("bootstrap.servers"));

        //接入协议。
        props.put(CommonClientConfigs.SECURITY_PROTOCOL_CONFIG, "SASL_PLAINTEXT");
        //Plain方式。
        props.put(SaslConfigs.SASL_MECHANISM, "PLAIN");
        //两次Poll之间的最大允许间隔。
        //消费者超过该值没有返回心跳，服务端判断消费者处于非存活状态，服务端将消费者从Consumer Group移除并触发Rebalance，默认30s。
        props.put(ConsumerConfig.SESSION_TIMEOUT_MS_CONFIG, 30000);
        //每次Poll的最大数量。
        //注意该值不要改得太大，如果Poll太多数据，而不能在下次Poll之前消费完，则会触发一次负载均衡，产生卡顿。
        props.put(ConsumerConfig.MAX_POLL_RECORDS_CONFIG, 30);
        //消息的反序列化方式。
        props.put(ConsumerConfig.KEY_DESERIALIZER_CLASS_CONFIG,
            "org.apache.kafka.common.serialization.StringDeserializer");
        props.put(ConsumerConfig.VALUE_DESERIALIZER_CLASS_CONFIG,
            "org.apache.kafka.common.serialization.StringDeserializer");
        //当前消费实例所属的消费组，请在控制台申请之后填写。
        //属于同一个组的消费实例，会负载均衡消费消息。
        props.put(ConsumerConfig.GROUP_ID_CONFIG, kafkaProperties.getProperty("group.id"));

        //消费offset的位置。注意！如果auto.offset.reset=none这样设置，消费组在第一次消费的时候就会报错找不到offset，第一次这时候就需要在
        props.put(ConsumerConfig.AUTO_OFFSET_RESET_CONFIG, "none");
        //构造消费对象，也即生成一个消费实例。
        KafkaConsumer<String, String> consumer = new KafkaConsumer<String, String>(props);
        //设置消费组订阅的Topic，可以订阅多个。
        //如果GROUP_ID_CONFIG是一样，则订阅的Topic也建议设置成一样。
        List<String> subscribedTopics = new ArrayList<String>();
        //如果需要订阅多个Topic，则在这里添加进去即可。
        //每个Topic需要先在控制台进行创建。
        String topicStr = kafkaProperties.getProperty("topic");
        String[] topics = topicStr.split(",");
        for (String topic : topics) {
            subscribedTopics.add(topic.trim());
        }
        consumer.subscribe(subscribedTopics);
        //循环消费消息。
        while (true) {
            try {
                ConsumerRecords<String, String> records = consumer.poll(1000);
```

```

//必须在下次Poll之前消费完这些数据,且总耗时不得超过SESSION_TIMEOUT_MS_CONFIG。建议开一个单独的线程池来消费消息,然后
for (ConsumerRecord<String, String> record : records) {
    System.out.println(
        String.format("Consume partition:%d offset:%d", record.partition(), record.offset());
    )
}
} catch (NoOffsetForPartitionException e) {
    System.out.println(e.getMessage());
}

//当auto.offset.reset设置为 none时,需要捕获异常 自己设置offset。您可以根据自身业务情况选择以下方式中的其中一种。
//e.g 1 :指定offset,这里需要自己维护offset,方便重试。
Map<Integer, Long> partitionBeginOffsetMap = getPartitionOffset(consumer, topicStr, true);
Map<Integer, Long> partitionEndOffsetMap = getPartitionOffset(consumer, topicStr, false);
consumer.seek(new TopicPartition(topicStr, 0), 0);

//e.g 2:从头开始消费
consumer.seekToBeginning(Lists.newArrayList(new TopicPartition(topicStr, 0)));

//e.g 3:指定offset为最近可用的offset。
consumer.seekToEnd(Lists.newArrayList(new TopicPartition(topicStr, 0)));

//e.g 4: 根据时间戳获取offset,就是根据时间戳去设置offset。例如重置到10分钟前的offset
Map<TopicPartition, Long> timestampsToSearch = new HashMap<>();
Long value = Instant.now().minus(300, ChronoUnit.SECONDS).toEpochMilli();
timestampsToSearch.put(new TopicPartition(topicStr, 0), value);
Map<TopicPartition, OffsetAndTimestamp> topicPartitionOffsetAndTimestampMap = consumer
    .offsetsForTimes(timestampsToSearch);
for (Entry<TopicPartition, OffsetAndTimestamp> entry : topicPartitionOffsetAndTimestampMap
    .entrySet()) {
    TopicPartition topicPartition = entry.getKey();
    OffsetAndTimestamp entryValue = entry.getValue();
    consumer.seek(topicPartition, entryValue.offset()); // 指定offset,这里需要自己维护offset,方便重试。
}
}
}
}

/**
 * 获取topic的最早、最近的offset
 * @param consumer
 * @param topicStr
 * @param beginOrEnd true begin; false end
 * @return
 */
private static Map<Integer, Long> getPartitionOffset(KafkaConsumer<String, String> consumer, String topicStr,
    boolean beginOrEnd) {
    Collection<PartitionInfo> partitionInfos = consumer.partitionsFor(topicStr);
    List<TopicPartition> tp = new ArrayList<>();
    Map<Integer, Long> map = new HashMap<>();
    partitionInfos.forEach(str -> tp.add(new TopicPartition(topicStr, str.partition())));
    Map<TopicPartition, Long> topicPartitionLongMap;
    if (beginOrEnd) {
        topicPartitionLongMap = consumer.beginningOffsets(tp);
    } else {
        topicPartitionLongMap = consumer.endOffsets(tp);
    }
    topicPartitionLongMap.forEach((key, beginOffset) -> {
        int partition = key.partition();
        map.put(partition, beginOffset);
    });
    return map;
}
}
}
}

```

**监控告警**  
**查看监控信息**

Last updated: 2023-04-11 09:54:28

## 操作场景

消息队列CKafka支持监控您账户下创建的资源，包括实例、Topic、Consumer Group等，帮助您实时掌握资源状态，针对可能存在的问题及时处理，保障其稳定运行。

本文为您介绍通过 CKafka 控制台查看监控指标的操作方法和监控指标的含义。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在实例列表中，单击需要查看的“实例 ID/名称”，进入实例详情页。
3. 在实例详情页顶部，单击**监控**，选择要查看的实例资源标签，设置好时间范围，可以查看实例监控数据。

图标	说明
	单击可查看监控指标同环比。
	单击可刷新获取最新的监控数据。
	单击可将图表复制到 Dashboard，关于 Dashboard 请参见 <a href="#">什么是 Dashboard</a> 。
<input checked="" type="checkbox"/> 显示图例	勾选后可在图表上显示图例信息。

## 监控信息展示

### 说明

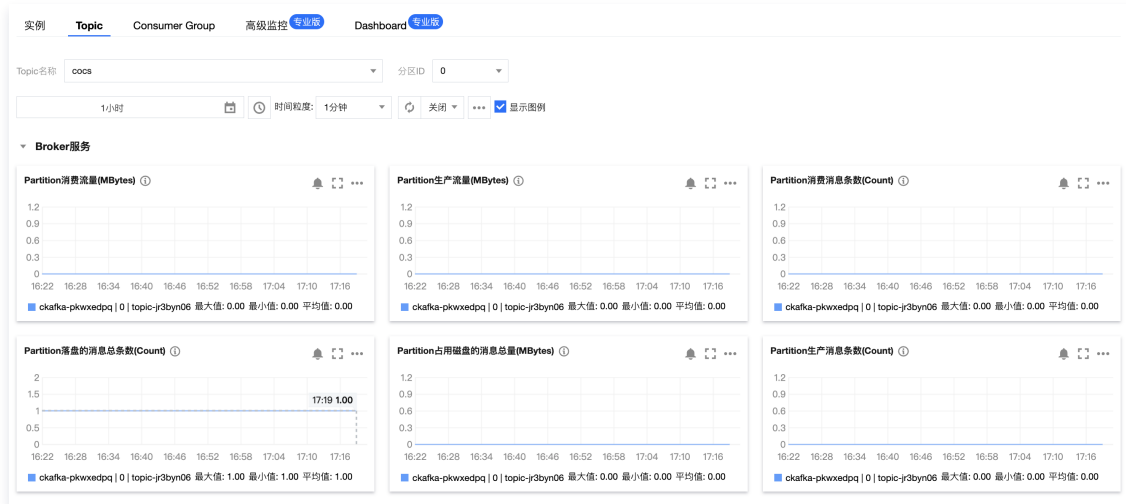
您可以单击以下页签，查看实例、Topic、Consumer Group 监控信息的具体展示。

### 实例

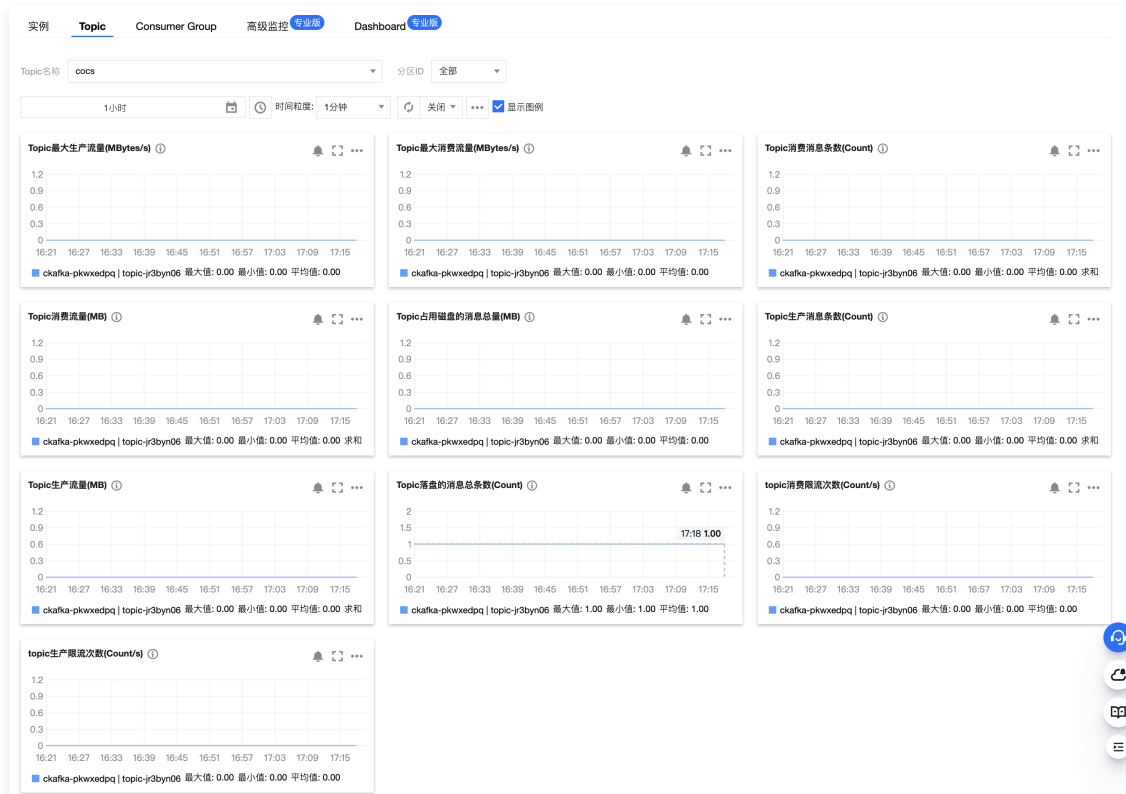


### Topic

选择分区后，可以查看指定 Partition 的监控数据。



不选择时默认全部，展示现有的 Topic 级别的监控数据。

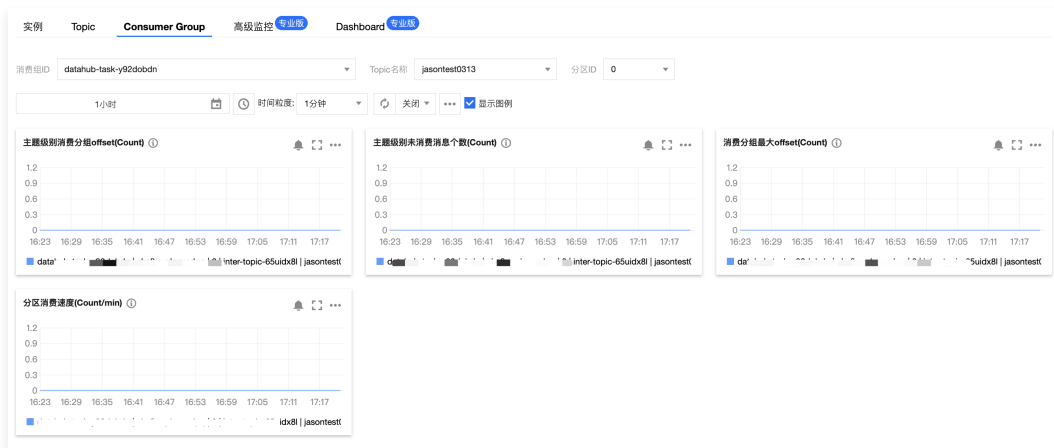


### Consumer Group

### Consumer Group-Topic 监控



Consumer Group-Partition 监控



监控指标含义及说明

以下是 CKafka 监控指标相关说明。其中备注**推荐**字样的指标是根据历史用户反馈统计建议对其配置监控和告警策略，请您根据实际情况合理配置监控和告警策略。

实例

监控指标	说明
实例磁盘占用量 (MB)	实例磁盘占用量（包含副本），按照所选择的时间粒度取最新值。
磁盘使用百分比 (%) ( <b>推荐</b> )	当前磁盘占用与实例规格磁盘总容量的百分比。
实例连接数 (Count) ( <b>推荐</b> )	客户端和服务器的连接数。
实例最大生产流量 (MB) ( <b>推荐</b> )	实例生产消息峰值带宽，不包含副本生产的带宽。（计算实例生产带宽使用百分比时的参考依据）。
实例最大消费流量 (MB) ( <b>推荐</b> )	实例消费消息峰值带宽，消费时无副本的概念。（计算实例消费带宽使用百分比时的参考依据）。
实例消费带宽百分比(%)	实例消费带宽占配额百分比
实例消费限流次数 (Count)	实例消费限流次数。
实例生产带宽百分比(%)	实例生产带宽占配额百分比。
实例生产限流次数 (Count)	实例生产限流次数。
实例落盘的消息总条数 (Count)	实例落盘的消息总条数（不包含副本），按照所选择的时间粒度取最新值。
实例消费流量 (MB)	实例消费流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。

实例生产消息条数 (Count)	实例生产消息条数，按照所选择的时间粒度统计求和。
实例生产流量 (MB)	实例生产流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。
实例消费消息条数 (Count)	实例消费消息条数，按照所选择的时间粒度统计求和。

## Topic

## Topic 监控

监控指标	说明
最大生产流量 (MB/s)	Topic 最大生产流量(不含副本流量)。
最大消费流量 (MB/s)	Topic 最大消费流量。
消费消息条数 (条)	Topic 的实际消费消息条数，按照所选择的时间粒度统计求和。
消费流量 (MB)	Topic 的实际消费流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。
占用磁盘的消息总量 (MB) (推荐)	Topic 实际占用磁盘的消息总量（不包含副本），按照所选择的时间粒度取最新值。
生产消息条数 (条)	Topic 的实际生产消息条数，按照所选择的时间粒度统计求和。
生产流量 (MB)	Topic 的实际生产流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。
落盘的消息总条数 (条)	Topic 的实际的落盘的消息总条数（不包含副本），按照所选择的时间粒度取最新值。
topic 消费限流次数(Count/s)	Topic 消费限流次数，按照所选择的时间粒度统计求和。
topic 生产限流次数(Count/s)	Topic 生产限流次数，按照所选择的时间粒度统计求和。

## Topic-Partition 监控

监控指标	说明
消费消息条数 (Count)	Partition 的实际消费消息条数，按照所选择的时间粒度统计求和。
消费流量 (MB)	Partition 的实际消费流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。
落盘的消息总条数 (Count)	Partition 的实际的落盘的消息总条数（不包含副本），按照所选择的时间粒度取最新值。
占用磁盘的消息总量 (MB)	Partition 实际占用磁盘的消息总量（不包含副本），按照所选择的时间粒度取最新值。
生产消息条数 (Count)	Partition 的实际生产消息条数，按照所选择的时间粒度统计求和。
生产流量 (MB)	Partition 的实际生产流量（不包含副本产生的流量），按照所选择的时间粒度统计求和。

## Consumer Group

## Consumer Group-Topic 监控

监控指标	说明
主题最大 offset	当前主题中所有分区的最大 offset。
主题消费 offset (区分 offset 最大值)	当前主题所有分区中消费组消费的最大 offset。
主题未消费消息数 (条)	当前主题所有分区消费组未消费消息总和。
主题消费速度 (条/分钟)	当前主题所有分区消费组消费速率总和。

**Consumer Group-Partition 监控**

监控指标	说明
分区消费速度（条/分钟）	消费分组在该分区的消费速率（条/分钟）。
当前消费 offset	消费分组该分区当前消费 offset。
当前分区最大 offset	当前 分区 最大 offset。
未消费的消息条数（条）（推荐）	消费分组在该分区下未消费消息数。

指标中英文对照详情参见文档 [CKafka 监控指标](#)。

**CKafka 监控 API 文档**

CKafka 监控 API 详情可参见如下文档：

- [主题监控](#)
- [实例监控](#)
- [消费分组监控](#)

# 查询高级监控（专业版）

Last updated: 2023-04-11 09:54:28

## 操作场景

CKafka 专业版支持高级监控功能，您可以在控制台查看核心服务、生产、消费和 Broker GC 等指标，方便运维人员在使用 CKafka 时进行排障处理。本章节为您介绍如何在控制台查看高级监控指标，并说明高级监控指标含义。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在实例列表中，单击需要查看的“实例 ID/名称”，进入实例详情页。
3. 在实例详情页顶部，单击 **监控** > **高级监控**，选择要查看的指标页签，设置好时间范围，可以查看监控数据。

## 监控信息展示

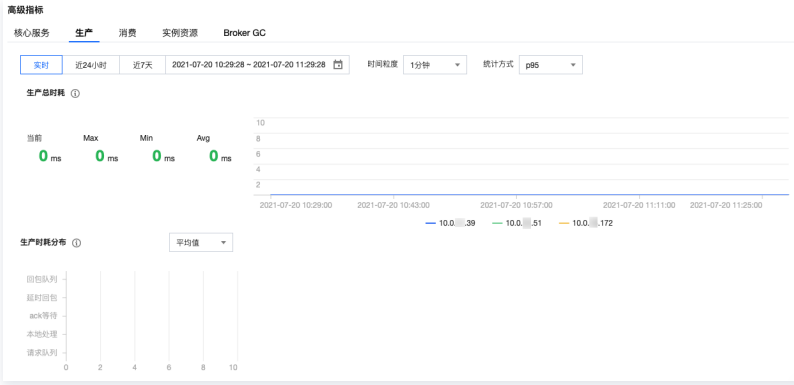
### 说明

您可以单击以下页签，查看核心服务、生产、消费、实例资源和 Broker GC 的监控信息的具体展示。

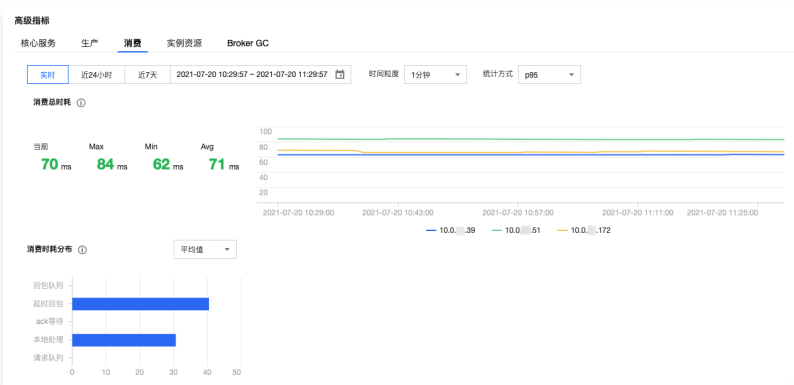
### 核心服务



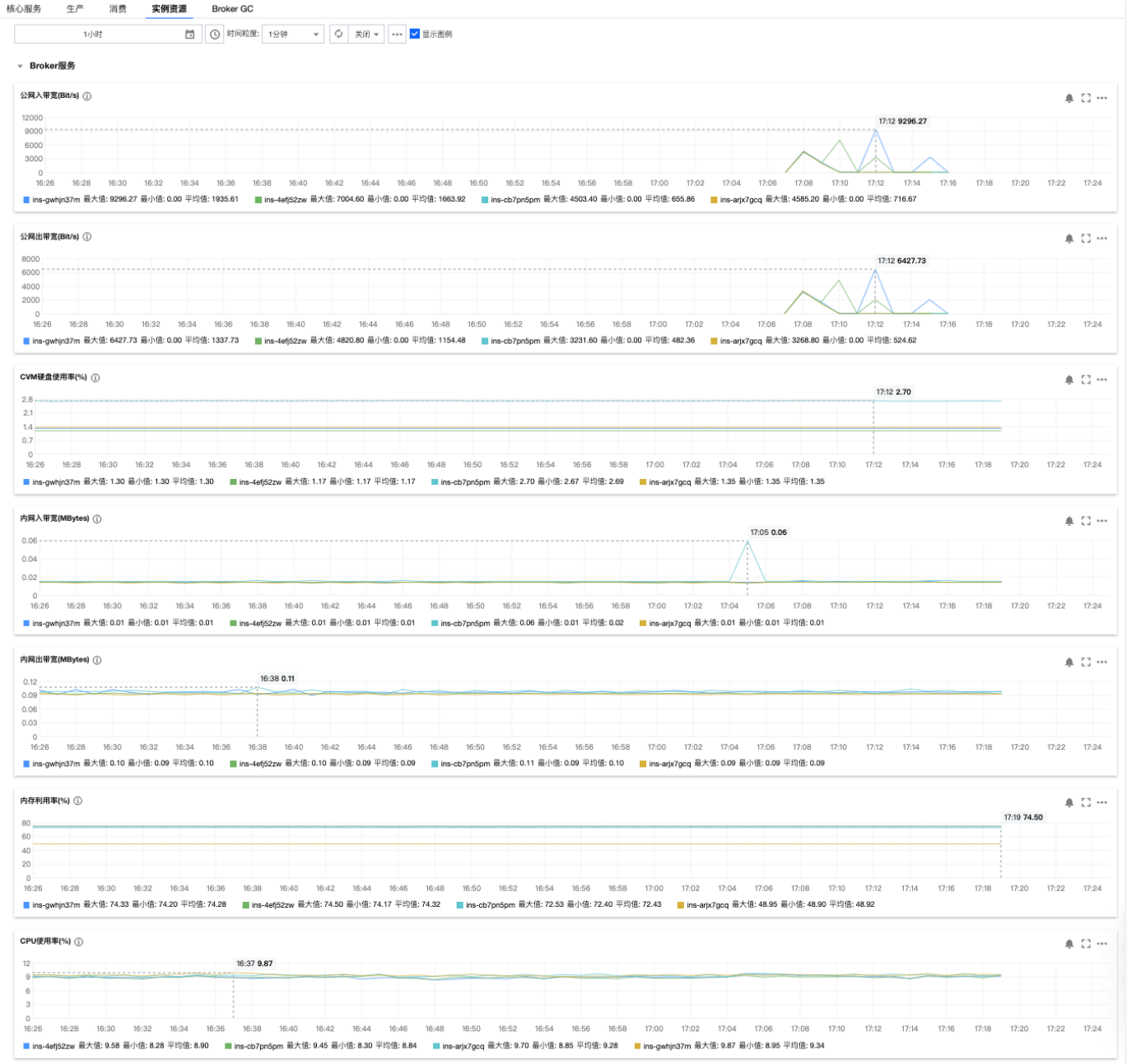
### 生产



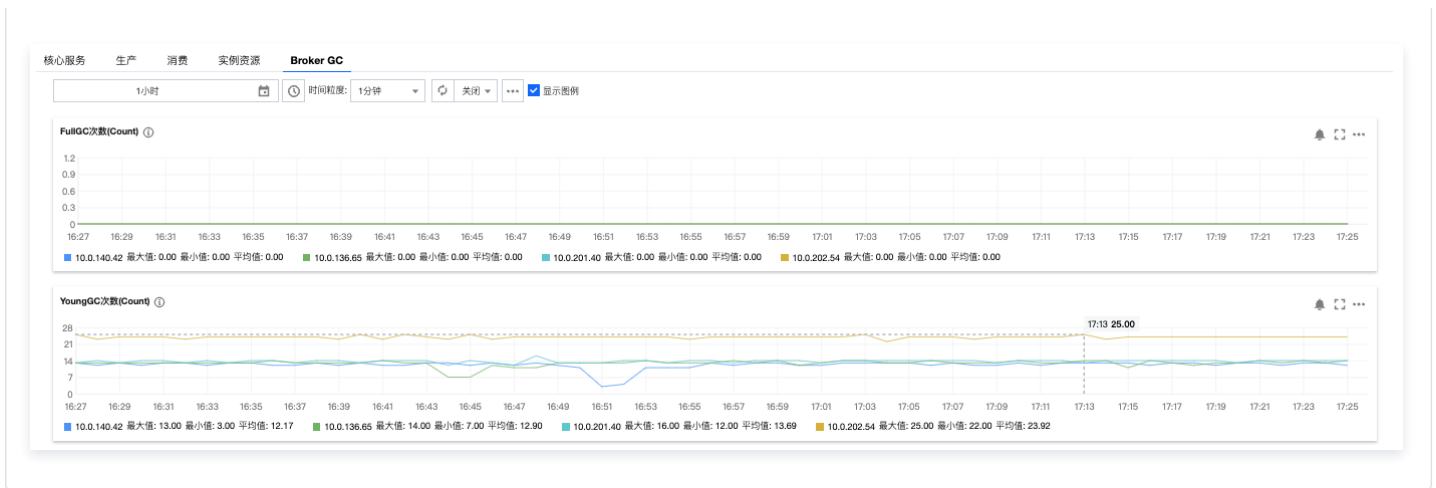
### 消费



### 实例资源



Broker GC



### 监控指标说明

#### 说明

您可以单击以下页签，查看核心服务、生产、消费、实例资源和 Broker GC 的监控指标的详细说明。

#### 核心服务监控

监控指标	指标说明	正常值范围
网络繁忙程度	用于衡量实例当前网络并发处理剩余的 IO 资源，越接近1越空闲。	该值一般在0.5 - 1之间波动，小于0.3表示负载较高。
请求队列深度	反映当前未处理的生产请求个数，如果该值过大可能是同一时间请求量过大，CPU 负载过高或者磁盘 IO 出现瓶颈。	<ul style="list-style-type: none"> <li>如果持续等于2000，表示集群负载比较高。</li> <li>小于2000的时候可忽略。</li> </ul>
未同步副本数	集群中存在的未同步的副本个数，当实例存在未同步副本，表示集群的健康度可能存在问题。	<ul style="list-style-type: none"> <li>曲线如果长期大于5（原因是有一些腾讯云内置的 Topic 的分区可能会处于离线状态，跟业务无关），表示需要处理集群。</li> <li>偶尔 Broker 波动，值凸起后，一段时间后，又回归平稳，属于正常现象。</li> </ul>
ZK 断连次数	Broker 和 Zookeeper之间的长连接断开重连的次数。网络波动，集群负载较高有可能会引起连接断开和重连，发生时会发生 leader 切换。	<ul style="list-style-type: none"> <li>无正常值范围。</li> <li>ZK 断连次数是累加的，次数越大不表示集群有问题，指标仅供参考。</li> </ul>
ISR 扩充次数	Kafka ISR 扩充次数，即存在未同步副本的情况下，当未同步副本追上 leader 数据，会重新加入 ISR，此时该次数就会加1。	<ul style="list-style-type: none"> <li>无正常值范围，当集群出现波动时，会出现扩充。</li> <li>如果非长时间大于0，无需关注。</li> </ul>
ISR 缩小次数	Kafka ISR 收缩次数，即当出现 Broker 宕机，Zookeeper 重连的情况，会出现 ISR 缩小的次数统计。	<ul style="list-style-type: none"> <li>无正常值范围，当集群出现波动时，会出现收缩。</li> <li>如果非长时间大于0，无需关注。</li> </ul>

#### 生产

监控指标	指标说明	正常值范围
生产总	表示生产请求的总耗时，由请求队列耗时，本地耗时，延时回包耗时等	<ul style="list-style-type: none"> <li>一般值在0 - 100ms之间，数据量大的时候0 - 1000ms</li> </ul>

时耗	指标汇总而成。 在每一个时间点，总耗时不等于以下五个耗时的累加，因为每个指标都是各自取平均得到的，故不累加相等。	的耗时属于正常范围。 <ul style="list-style-type: none"> <li>只要不长时间大于1000ms，可以不用处理。</li> </ul>
请求队列耗时	生产请求在接收请求队列内等待的时间，表示请求包等待后续的进程处理。	<ul style="list-style-type: none"> <li>一般值在0 - 50ms之间，数据量大的时候0 - 200ms的耗时属于正常范围。</li> <li>只要不长时间大于200ms，可以不用处理。</li> </ul>
本地处理耗时	生产请求在 Leader Broker 处理的时间，即从请求队列拿出请求包，写入到本地 page cache 的时间。	<ul style="list-style-type: none"> <li>一般值在0 - 50ms之间，数据量大的时候0 - 200ms的耗时属于正常范围。</li> <li>只要不长时间大于200ms，可以不用处理。</li> </ul>
ack 等待耗时	生产请求等待数据同步的耗时，当客户端 ack = -1时该值才会大于0，即只要 ack = 1或者0，该值都是0。	<ul style="list-style-type: none"> <li>一般值在0 - 200ms之间，数据量大的时候0 - 500ms的耗时属于正常范围。</li> <li>只要不长时间大于500ms，可以不用处理。</li> <li>跨可用区的实例当 ack = -1时，该值会比非跨可用区的实例高，具体参见 <a href="#">跨可用区部署</a>。</li> </ul>
延时回包耗时	生产请求被系统延时回包的耗时，当实例的流量没超过购买流量时，该值都为0，如果被限流，则会大于0。	<ul style="list-style-type: none"> <li>当实例不超限时，该值长期为0。</li> <li>如果实例超限，会根据超限的比例，延时0 - 5分钟，即该值最大为5分钟。</li> </ul>
回包队列耗时	生产请求在回包队列的等待耗时，表示请求包在等待被发送给客户端。	<ul style="list-style-type: none"> <li>一般值在0 - 50ms之间，数据量大的时候0 - 200ms的耗时属于正常范围。</li> <li>只要不长时间大于200ms，可以不用处理。</li> </ul>

消费

监控指标	指标说明	正常值范围
消费总耗时	表示消费的总耗时，由请求队列耗时，本地耗时等指标汇总而成。 在每一个时间点，总耗时不等于以下五个耗时的累加，因为每个指标都是各自取平均得到的，故不累加相等。	一般值会在于500ms - 1000ms之间（客户端默认的 fetch.max.wait.ms=500ms），数据量大的时候500 - 5000ms的耗时属于正常范围。
请求队列耗时	消费请求在请求队列内等待的时间，表示请求包等待后续的进程处理。	<ul style="list-style-type: none"> <li>一般值在0 - 50ms之间，数据量大的时候0 - 200ms的耗时属于正常范围。</li> <li>只要不长时间大于200ms，可以不用处理。</li> </ul>
本地处理耗时	消费请求在 Leader Broker 拉取数据的时间，即从本地磁盘读取数据。	<ul style="list-style-type: none"> <li>一般值在0 - 500ms之间，数据量大的时候0 - 1000ms的耗时属于正常范围。</li> <li>只要不长时间大于1000ms，可以不用处理，因为消费有可能会有读冷数据的情况，此时耗时会较高。</li> </ul>
消费等待耗时	客户端默认的 fetch.max.wait.ms = 500ms，即表示客户端允许在读不到数据的时候，服务端等待多久返回包给客户端。	一般值会在于500ms左右（客户端默认的 fetch.max.wait.ms = 500ms），该值的大小取决于客户端的参数设置。
延时回包耗时	消费请求被系统延时回包的耗时，当实例的流量没超过购买流量时，该值都为0，如果被限流，则会大于0。	<ul style="list-style-type: none"> <li>当实例不超限时，该值长期为0。</li> <li>如果实例超限，会根据超限的比例，延时0 - 5分钟，即该值最大为5分钟。</li> </ul>
回包队列耗时	消费请求在回包队列的等待耗时，表示请求包在等待被发送给客户端。	<ul style="list-style-type: none"> <li>一般值会在于0 - 50ms之间，数据量大的时候0 - 200ms的耗时属于正常。</li> <li>只要不长时间大于200ms，就可以不用处理。</li> </ul>

## 实例资源

监控指标	指标说明	正常值范围
CPU 使用率 (%)	CPU 使用率是某进程在一段时间内占用的 CPU 时间占总的 CPU 时间的百分比。	<ul style="list-style-type: none"> <li>该值一般会在 1-100 之间。</li> <li>持续5个以上周期范围内在90以上，则表示系统负载很高，需要处理排查。</li> </ul>
磁盘使用率 (%)	CVM 云服务器挂载硬盘的使用情况	<ul style="list-style-type: none"> <li>该值一般会在 0-100之间。</li> <li>如果该值达到80以上，则需要扩容处理。</li> </ul>
内网入带宽 (MB)	内网入带宽是指 CVM 在集群内通信能够达到的带宽，根据不同的规格限制内网带宽和内网收包能力。	<ul style="list-style-type: none"> <li>该值一般会大于0（集群内 CVM 监控会有数据产生）。</li> <li>如果没有入带宽产生则代表 CVM 服务异常，或网络异常不可到达。</li> </ul>
内网出带宽 (MB)	内网出带宽是指CVM在集群内通信能够达到的带宽，根据不同的规格限制内网带宽和内网发包能力。	<ul style="list-style-type: none"> <li>该值一般会大于0（集群内CVM监控会有数据产生）。</li> <li>如果没有出带宽产生则代表 CVM 服务异常，或网络异常不可到达。</li> </ul>
内存利用率 (%)	内存总空间减去所有内存可用空间占用内存总空间的百分比为内存利用率	<ul style="list-style-type: none"> <li>正常取值范围1-100。</li> <li>如果内存使用率达到90以上，则代表程序占用内存过高，需要对某些进程处理。</li> </ul>
公网入带宽 (MB)	公网入带宽是指CVM在公网通信能达到的带宽，根据不同规格限制公网带宽和外网收包能力。	有公网入流量下会大于0，否则为0。
公网出带宽 (MB)	公网出带宽是指CVM在公网通信能达到的带宽，根据不同规格限制公网带宽和外网发包能力。	有公网出流量下会大于0，否则为0。

## Broker GC

监控指标	指标说明	正常值范围
年轻代回收次数	Broker Yong GC 的次数	<ul style="list-style-type: none"> <li>正常在 0 - 300之间。</li> <li>如果持续高于300，则需要调整 GC 参数。</li> </ul>
老年代回收次数	Broker Full GC 的次数	<ul style="list-style-type: none"> <li>正常为0。</li> <li>如果有出现大于0的情况，则需要处理。</li> </ul>

## 监控指标异常原因

以下列举了部分监控指标异常原因说明。

指标	异常原因说明
CPU 使用率 (%)	当发现持续5个周期以上，使用率达到90%以上，可先确认下是否存在消息压缩和消息格式转换。如果客户端机器 CPU 资源很充足那么建议开启 snappy 压缩。可同时观察请求队列深度，如果该值过大可能是同一时间请求量过大，也可以造成 CPU 负载过高。
未同步副本数 (Count)	当出现大于 0 的个数时，即代表集群中存在的未同步的副本个数，当实例存在未同步副本，通常情况下是由于 Broker 节点异常或网络因素，可通过 Broker 日志来排查原因。
FullGC 次数 (Count)	当偶尔出现一次情况，很可能是磁盘 IO 导致，可能云服务器有关，可以关注下后续是否还有相同机器告警，如果相同 IP 机器

	经常出现问题，请联系 <a href="#">在线客服</a> 。
请求队列深度 (Count)	客户端生产消费出现超时而云服务器负载却正常的情况下，说明单台服务器的请求队列已达到最大上限，默认单台节点 broker 的请求队列长度是默认的配置是 500，可根据购买的资源配置情况适当调整（联系 <a href="#">在线客服</a> ）。

## 配置告警

Last updated: 2023-08-22 14:16:31

### 操作场景

腾讯云可观测平台产品默认为用户提供监控功能，无需用户手动开通。用户在使用了腾讯云某个产品后，可观测平台才可以开始收集监控数据。

消息队列 CKafka 支持监控您账户下创建的资源，包括实例、Topic、Consumer Group，帮助您实时掌握资源状态。您可以为监控指标配置告警规则，当监控指标达到设定的报警阈值时，可观测平台可以通过邮件、短信、微信、电话等方式通知您，帮助您及时应对异常情况。

### 操作步骤

#### 配置告警规则

创建的告警会将一定周期内监控的指标与给定阈值的情况进行比对，从而判断是否需要触发相关通知。当CKafka 状态改变而导致告警触发后，您可以及时进行相应的预防或补救措施，合理地创建告警能帮助您提高应用程序的健壮性和可靠性。

##### ① 说明

请务必对实例配置告警，防止因突发流量或者到达规格限制而导致的异常。

1. 登录 [CKafka 控制台](#)。
2. 在实例列表中，单击操作列的**配置告警**可以直接跳转到告警配置页面。
3. 在告警策略页面，选择好策略类型和要设置告警的实例，设置好告警规则和告警通知模板。
  - **监控类型**：选择**云产品监控**。
  - **策略类型**：选择**消息服务 CKafka**。
  - **告警对象**：选择需要配置告警策略的 CKafka 资源。
  - **触发条件**：支持**选择模板**和**手动配置**，默认选择手动配置，手动配置参见以下说明，新建模板参见 [新建触发条件模板](#)。

##### ① 说明

- **指标**：例如“磁盘使用百分比”，选择统计粒度为1分钟，则在1分钟内，磁盘使用百分比连续N个数据点超过阈值，就会触发告警。
- **告警频次**：例如“每30分钟警告一次”，指每30分钟内，连续多个统计周期指标都超过了阈值，如果有一次告警，30分钟内就不会再次进行告警，直到下一个30分钟，如果指标依然超过阈值，才会再次告警。推荐配置告警策略的指标参见 [CKafka 推荐监控告警策略](#)。

- **通知模板**：选择通知模板，也可以新建通知模板，设置告警接收对象和接收渠道。

4. 单击**完成**，完成配置。

##### ① 说明

有关告警的更多信息，请参见 [新建告警策略](#)。

#### 新建触发条件模板

1. 在配置告警规则页面，触发条件点击**选择模板**，单击**新建触发条件模板**，
2. 在新建模板页，配置策略类型。
  - **策略类型**：选择**消息服务 CKafka**。
  - **使用预置触发条件**：勾选此选项，会出现系统建议的告警策略。
3. 确认无误后，单击**保存**。

新建
✕

① 尊敬的用户您好，云监控事件告警创建入口已下线，存量事件告警计划于2022年4月中旬停止服务，相关能力将由事件总线承载，并在原有功能上新增规则匹配、自定义事件集、多目标投递等特性。为保证您的事件相关服务可以正常使用，我们建议您开通事件总线并进行能力迁移，同时也提供一键迁移服务，如果您有疑问可查看事件总线产品文档。

模板名称

备注

策略类型 消息服务CKafka-实例  使用预置触发条件①

触发条件  指标告警

满足 任意 条件时，触发告警

if	磁盘使用百分比	统计周期1分钟	>	80	%	持续1个周期	then	每1天警告一次	① ✕
if	实例连接数百分比	统计周期1分钟	>	80	%	持续1个周期	then	每1天警告一次	① ✕
if	实例生产带宽百分比	统计周期1分钟	>	80	%	持续1个周期	then	每1天警告一次	① ✕
if	实例消费带宽百分比	统计周期1分钟	>	80	%	持续1个周期	then	每1天警告一次	① ✕

[添加](#)

保存
取消

4. 返回新建告警策略页，单击刷新，就会出现刚配置的告警策略模板。

触发条件  选择模板  手动配置 (事件相关告警信息暂不支持通过触发条件模板配置)

ckafka实例告警 如无适合模板，您可以 [新增触发条件模板](#) 或 [修改模板](#)

**指标告警**

满足以下 任意 指标判断条件时，触发告警

if	磁盘使用百分比	统计粒度1分钟	>	80	%	持续5个数据点	then	每30分钟告警一次	①
if	实例连接数百分比	统计粒度1分钟	>	80	%	持续5个数据点	then	每30分钟告警一次	①
if	实例生产带宽百...	统计粒度1分钟	>	80	%	持续5个数据点	then	每30分钟告警一次	①
if	实例消费带宽百...	统计粒度1分钟	>	80	%	持续5个数据点	then	每30分钟告警一次	①

### CKafka 推荐监控告警策略

影响业务数据稳定性的指标请参见 [CKafka 数据可靠性说明](#)。

根据历史用户反馈统计，建议对 CKafka 以下3个维度（共6项指标）配置告警策略，请您根据实际情况合理配置告警策略。

**实例监控：**

监控指标	说明
生产峰值带宽（MB/s）	实例生产消息时产生的最大流量（不包含副本产生的流量）。
消费峰值带宽（MB/s）	实例消费消息时产生的最大流量（消费时无副本的概念）。
磁盘使用百分比（%）	当前磁盘占用与实例规格磁盘总容量的百分比。
实例连接数（个）	客户端与服务器的连接数。

**Topic 监控：**

监控指标	说明
已占用磁盘容量（MB）	Topic 实际占用磁盘的消息总量（不包含副本），按照所选择的时间粒度取最新值。

## 消费分组:

监控指标	说明
未消费的消息条数 ( 条 )	消费分组未消费消息数。

## 告警配置建议

Last updated: 2022-02-08 11:10:10

消息队列 CKafka 不仅为运行中的 CKafka 集群提供了多项监控指标，用于监测集群的运行情况，还提供了一些关键指标的配置告警功能，帮助您及时发现集群问题并进行处理。具体使用方法可参见 [查看监控](#) 和 [配置告警](#)。

本文为您介绍在使用 CKafka 过程中需要重点关注的一些指标及其告警建议配置：

指标	告警建议配置	详细说明
磁盘使用率 (%)	统计周期1分钟，>80%，持续5个周期，每30分钟告警一次	平均磁盘使用率表示集群各节点磁盘使用率的平均值。磁盘使用率过高会导致节点没有足够的磁盘空间容纳分配到该节点上，从而导致消息无法落盘，建议在平均磁盘使用率超过75%时及时清理数据或扩容集群。
未消费的消息条数 (Count)	统计周期5分钟，>8000，持续10个周期，每30分钟告警一次	堆积过多的消息会导致 Broker 节点磁盘使用率迅速上涨，无法再接入更多消息，服务会停止。需要进行扩容
生产峰值带宽 (MB/s)	统计周期1分钟，>所购买的实例带宽规格，持续5个周期，每10分钟告警一次	一分钟内，客户每秒的流量最大值。判断是否超出当前所购买的流量上限。可根据此项适当选择升配操作等

## 查看高级运维特性（专业版）

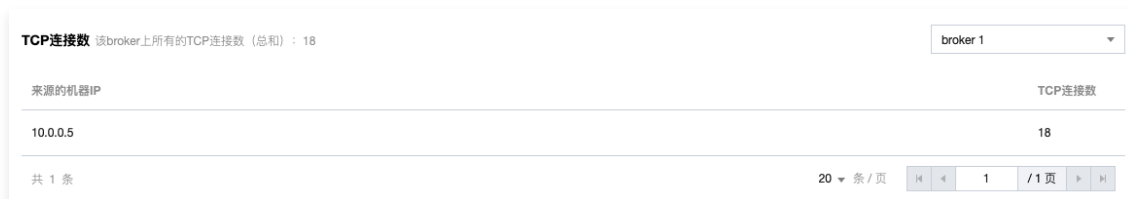
Last updated: 2023-04-11 09:54:29

### 操作场景

CKafka 专业版支持高级运维特性，您可以在控制台查看 TCP 连接数、未同步副本详情、Topic/Consumer Group 统计排行等，方便运维人员在使用 CKafka 时进行排障处理。

### 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在实例列表中，选择好地域，单击需要查看的“实例 ID/名称”，进入实例详情页。
3. 在实例详情页顶部，单击**监控 > Dashboard**，设置好时间范围，查看相关排行信息。
  - TCP 连接数：展示该 broker 上所有的 TCP 连接数（总和），实例连接数将满时，便于用户查看各个机器的连接数情况。



来源的机器IP	TCP连接数
10.0.0.5	18

- 未同步副本详情：未同步消息的副本详情。



Topic ID	Topic名称	分区名称	未同步副本
暂无数据			

- 统计排行：
  - Topic：展示 Topic 生产消费流量 Top10 和 占用磁盘容量 Top10。

统计排行

Topic Consumer Group

10分钟以内 近24小时 近7天 近30天 2023-04-10 16:59:58 - 2023-04-10 17:59:58

Topic流量 Top 10 生产

Topic ID	Topic名称	分区数	副本数	Topic 流量(MB/min)	Topic 数据堆积(不含副本)(MB)
1	topic-no-aci	3	2	0.00	0
2	topic-test0313	1	2	0.00	0
3	topic-test44	1	2	0.00	0
4	topic-topic_100_3	100	1	0.00	0
5	topic-test1	2	2	0.00	0
6	inter-test-egg2	10	2	0.00	0
7	topic-testacl	1	2	0.00	0
8	topic-fctracker_uni_v2	80	1	0.00	0
9	topic-topic_acl	3	2	0.00	0
10	inter-test-jasontest0313	1	3	0.00	0

Topic占用磁盘容量 Top 10

Topic ID	Topic名称	分区数	副本数	Topic 流量(MB/min)	占用磁盘容量(MB)
1	topic-no-aci	3	2	0.00	0
2	topic-test0313	1	2	0.00	0
3	topic-test44	1	2	0.00	0
4	topic-topic_100_3	100	1	0.00	0
5	topic-test1	2	2	0.00	0
6	inter-test-egg2	10	2	0.00	0
7	topic-testacl	1	2	0.00	0
8	topic-fctracker_uni_v2	80	1	0.00	0
9	topic-topic_acl	3	2	0.00	0
10	inter-test-jasontest0313	1	3	0.00	0

○ Consumer Group: 展示 Consumer Group 消费速度 Top10。

统计排行

Topic Consumer Group

10分钟以内 近24小时 近7天 近30天 2023-04-10 16:59:58 - 2023-04-10 17:59:58

Consumer Group消费速度 Top 10

消费组名称	消费速度(条/min)	操作
1 \consumer\	0	<a href="#">查看监控</a>
2 \test\	0	<a href="#">查看监控</a>
3 cg1	0	<a href="#">查看监控</a>
4 cg1	0	<a href="#">查看监控</a>
5 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>
6 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>
7 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>
8 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>
9 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>
10 ckafka-pkwxedpq_ckafka-vvj7xem4	0	<a href="#">查看监控</a>

# 智能运维

## 一键诊断

Last updated: 2022-06-21 17:57:01

CKafka 推出一键诊断功能，该功能能够主动排查集群问题和隐患，并基于腾讯云专家经验沉淀给出问题解决方案，自动归纳健康检查结果生成报告。一键诊断能力能够为用户提取关键信息、定位问题、提供专业解决建议，实现运维体验闭环。

### 说明

当前仅 CKafka 专业版实例支持一键诊断功能，且每天只能诊断一次，请您合理使用诊断次数。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例列表**，单击目标实例的“ID”，进入实例基本信息页面。
3. 在页面上方选择**智能运维**页签，二级页签选择**一键诊断**。
4. 在一键诊断页面左上角单击**一键诊断**按钮，开始诊断，诊断结束后诊断结果将会展示在页面上。您也可以开启**自动诊断**，开启后，每天会在指定时间主动触发一次诊断，并生成一份诊断报告。



### 查看诊断结果

在诊断结果页面，您可以进行以下操作：

- 当诊断结论异常时，您可以点击 **最新诊断结果** 模块左下角的 **查看诊断报告**，根据诊断报告中针对风险项的详情说明和诊断建议及时对问题进行修复。
- 单击**诊断报告**模块右上角的 按钮，可下载诊断报告。
- 单击**诊断报告**模块右边栏的时间，可查看历史诊断记录的诊断报告。

诊断报告

集群诊断报告



诊断时间 2021-10-28 21:56:15

实例ID ckafka-██████████

诊断结论 实例存在高风险诊断项1个，低风险诊断项3个。已经影响或即将影响实例的可用性，建议您尽快查看和处理。

▼ 高风险诊断项: 1

🚨 网络繁忙程度

诊断项说明 用于衡量实例当前网络线程处理能力的指标，越接近1越空闲

诊断结果 1

诊断建议 您当前集群的网络繁忙程度指标过高,请检查网络繁忙程度指标监控,如影响到了业务的生产消费请提工单反馈处理。

▼ 低风险诊断项: 3

🚩 zk断连次数

诊断项说明 Broker 和 Zookeeper之间的长连接断开重连的次数。网络波动，集群负载较高有可能会引起连接断开&重连。发生时会发生leader切换。该值是一个累加值，Broker启动后，断连一次加1，只有broker重启才会置0

诊断结果 0

诊断建议 您当前集群的zk断连次数指标过高,请检查zk断连次数指标监控,如影响到了业务的生产消费请提工单反馈处理。

🚩 ISR缩小抖动次数

诊断项说明 Kafka ISR 收缩次数，即当出现broker 宕机，zookeeper重连的情况，会出现ISR缩小的次数统计

诊断结果 0

诊断建议 您当前集群的ISR缩小抖动次数指标过高,请检查ISR缩小抖动次数指标监控,如影响到了业务的生产消费请提工单反馈处理。

🚩 ISR扩充次数

诊断项说明 Kafka ISR 扩充次数，即存在未同步副本的情况下，当未同步副本追上leader数据，会重新加入ISR，此时该次数就会加1

诊断结果 0

诊断建议 您当前集群的ISR扩充次数指标过高,请检查ISR扩充次数指标监控,如影响到了业务的生产消费请提工单反馈处理。

# 弹性伸缩

## 磁盘水位处理

Last updated: 2023-07-05 11:22:11

### 操作场景

消息队列 CKafka 支持磁盘水位自动调整功能，在磁盘水位到达阈值后，您可以选择设置**动态消息保留策略**来降低消息保留时间或者设置**磁盘自动扩容**来调整磁盘空间。

- **动态消息保留**：用户设置消息保留时间后，过期的消息就会被删除。若遇到用户消息猛增的情况，磁盘空间满了之后，则无法正常生产和消费。设置数据动态保留策略后，当磁盘空间使用率达到一定的比例后，会自动向前过期一定比例的数据，避免上述情况出现。
- **磁盘自动扩容**：当磁盘负载过重之后，会无法正常生产和消费，设置磁盘自动扩容策略后，若磁盘负载到达触发阈值，会自动根据扩容策略调整磁盘容量，避免上述情况出现。

策略类型	CKafka 标准版是否支持	CKafka 专业版是否支持
动态消息保留	是	是
磁盘自动扩容	否	是

#### 说明

动态消息保留策略和磁盘自动扩容最多只能有一个开启。

### 功能入口

#### 标准版

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在实例基本信息页面的**磁盘水位处理**模块，您可以设置磁盘水位处理策略调整磁盘空间。

#### 专业版

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在页面上方选择**智能运维**页签后，二级页签选择**弹性伸缩**，在磁盘水位处理模块，设置磁盘空间调整策略。

### 设置磁盘水位处理策略

#### 动态消息保留策略

1. 在**磁盘水位处理**模块，开启**动态保留策略**。

#### 说明

默认动态策略是当磁盘负载达到75%以后，向前调整消息保留时间10%。

## 磁盘水位处理②

策略类型	状态	策略描述	调整记录	操作
动态消息保留策略	<input type="checkbox"/>	当磁盘负载达到 75% 时，向前调整消息保留时间 10%。保底时长 6 小时	<a href="#">查看</a>	<a href="#">配置</a>
磁盘自动扩容	<input checked="" type="checkbox"/>	当磁盘负载达到 75% 时，磁盘自动扩容 10%。最大磁盘 500000GB	<a href="#">查看</a>	<a href="#">配置</a>

2. 单击动态保留策略操作列的**配置**，可设置动态策略和保底时长。

**动态保留策略** ×

动态策略 当磁盘负载达到  % 时

向前调整消息保留时间  %

触发调整消息保留时间后，Broker会根据新的保留时间删除部分最早的历史数据，该效果会有一定延时

保底时长  小时

范围：1 分钟到30 天，动态调整到该时间后不再触发动态调整

- 动态策略：触发调整消息保留时间后，Broker 会根据新的保留时间删除部分最早的历史数据，该效果会有一定延迟。
- 保底时长：1分钟-30小时，动态调整到该时间后不再触发动态调整。

3. 单击调整记录列的**查看**可跳转至事件中心查看磁盘自动扩容调整记录详情。

## 磁盘自动扩容策略

1. 在**磁盘水位处理**模块，开启**磁盘自动扩容**。

## ! 说明

默认当磁盘负载达到 75% 时，磁盘自动扩容 10%。最大磁盘是当前实例规格对应的磁盘上限，以控制台展示为准。

## 磁盘水位处理②

策略类型	状态	策略描述	调整记录	操作
动态消息保留策略	<input type="checkbox"/>	当磁盘负载达到 75% 时，向前调整消息保留时间 10%。保底时长 6 小时	<a href="#">查看</a>	<a href="#">配置</a>
磁盘自动扩容	<input checked="" type="checkbox"/>	当磁盘负载达到 75% 时，磁盘自动扩容 10%。最大磁盘 500000GB	<a href="#">查看</a>	<a href="#">配置</a>

2. 单击磁盘自动扩容操作列的**配置**，可设置动态策略和最高磁盘。

## 磁盘自动扩容策略



动态策略 当磁盘负载达到  % 时

磁盘自动扩容  %

磁盘扩容的步长为100GB

最高磁盘  GB

自动扩容到该磁盘规格后不再触发自动扩容

确定

取消

- 动态策略：磁盘负载到达触发阈值后，会自动根据扩容策略调整磁盘容量，该效果会有一定延迟。

**说明**

磁盘扩容的步长是100GB。

- 最高磁盘：自动扩容到该磁盘规格后不再触发自动扩容。

3. 单击调整记录列的查看可跳转至事件中心查看磁盘自动扩容调整记录详情。

# 分区动态处理

Last updated: 2023-04-11 09:54:29

## 操作场景

消息队列 CKafka 支持分区动态处理功能，当您开启**自动化分区均衡**后，CKafka 会根据您设定的时间自动检查 Topic 的分区分布，自动分析选择业务低峰期发起分区均衡。

### 说明

该功能仅 CKafka 专业版实例支持。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏单击**实例列表**，单击目标实例的“ID/名称”，进入基本信息页。
3. 在页面上方选择**智能运维**页签后，二级页签选择**弹性伸缩**。
4. 在弹性伸缩页面的**分区动态处理**模块，开启**自动化分区均衡**。



5. 单击自动化分区均衡操作列的**配置**，可设置自动化分区均衡策略。



- 自定义时间：自定义选择发起分区均衡的时间，建议选择业务低峰期，避免对您的业务造成影响。
- 自动分析选择业务低峰期：CKafka 根据自动分析得出的业务低峰期选择发起分区均衡的时间。

6. 单击调整记录栏的**查看**可跳转至事件中心查看自动化分区均衡调整记录详情。

# 权限管理

## 配置 ACL 策略

Last updated: 2023-06-29 09:38:32

### 操作场景

该任务指导您在使用消息队列 CKafka 时，通过控制台配置 SASL 鉴权和 ACL 规则，增强对公网/内网传输中的用户访问控制，增加对 Topic 等资源的生产消费权限控制。

#### 说明

- CKafka 提供了多种安全认证机制，主要分为 SSL 和 SASL2 大类，其中 SASL/PLAIN 是基于账号密码的认证方式，比较常用。CKafka 支持 SASL\_PLAINTEXT 认证（参见 [添加路由策略-公网域名接入](#)）。
- ACL 访问控制列表（Access Control List），帮助用户定义一组权限规则，允许/拒绝用户 user 通过 IP 读/写 Topic 资源。

### 操作步骤

#### 新建用户

- 登录 [CKafka 控制台](#)。
- 在顶部菜单栏，选择地域后，单击目标实例“ID/名称”。
- 在实例详情页面，单击顶部 **ACL 策略管理** 页签。
- 选择 **用户管理** 子页签，单击 **新建**，填写用户名和密码信息，创建用户。

#### 配置 ACL 策略

- 在 ACL 策略管理页面，选择子页签 **策略列表**。
- 单击 **批量配置**，为用户授予权限。

#### 说明

- 若只设置允许规则，则除允许的规则外的其他 IP 都无法连接实例。
- 若只设置拒绝规则，则除拒绝的规则外的其他 IP 需要设置了允许规则后才可以连接实例。
- 若同时设置允许规则和拒绝规则，则只有允许规则中的 IP 可以连接实例，其他 IP 都无法连接实例。

#### 2.4.1版本及以上实例

支持 **批量勾选**，按 **前缀模糊匹配** 和 **预设规则** 三种方式为用户授予权限。

#### 说明

配置 ACL 策略时支持输入多个 IP 或网段，用 ; 隔开，若 IP 为空，则默认为 **全部 IP** 添加权限。

- 批量勾选**：选择多个需要配置相同 ACL 策略的 Topic。
- 按前缀模糊**：按 Topic 名称前缀模糊匹配需要配置相同 ACL 策略的 Topic，需要指定模糊匹配规则名称。设置后，新增按指定前缀命名的 Topic 时，系统自动配置指定 ACL 策略。

#### 说明

模糊匹配规则最多支持设置五条。

### 新增ACL策略 ×

**ACL策略示例：** 允许/拒绝 用户 user 通过 ip 读/写 topic资源

匹配方式  批量勾选  按前缀模糊匹配  预设规则

规则名称

按前缀匹配  ✕ 🔍

匹配Topic

TopicId	Topic名
topic-██████	rxtest1
topic-██████	test3

ACL策略

操作权限	用户	IP或网段	策略
<input type="text" value="允许"/>	<input type="text" value="请选择, 不选默认全部"/>	<input type="text" value="请输入IP或网段, 支持“;”分隔, 默认*"/>	<input type="text" value="写"/>

[添加规则](#)

- **预设规则：** 预设一套策略，后续创建 Topic 的时候可以自动应用这套规则。

**说明**

预设规则最多支持设置五条。

## 新增ACL策略



ACL策略示例：允许/拒绝 用户 user 通过 ip 读/写 topic资源

匹配方式

批量勾选

按前缀模糊匹配

预设规则

规则名称

rule

ACL策略

操作权限	用户	IP或网段	策略
允许 ▾	请选择, 不选默认全部 ▾	请输入IP或网段, 支持“;”分隔, 默认*	写 ▾
<a href="#">添加规则</a>			

自动应用至后续所有新增topic



提交

关闭

## 其他版本实例

支持**批量勾选**和**预设规则**两种方式为用户授予权限。

ⓘ 说明

支持输入多个 IP 或网段, 用 ; 隔开; 若 IP 为空, 则默认为**全部 IP**添加权限。

- **批量勾选**: 选择多个需要配置相同 ACL 策略的 Topic。
- **预设规则**: 预设一套策略, 后续创建 Topic 的时候可以自动应用这套规则。

ⓘ 说明

预设规则最多支持设置五条。

### 新增ACL策略 ✕

i ACL策略示例：允许/拒绝 用户 user 通过 ip 读/写 topic资源

匹配方式 批量勾选 预设规则

规则名称 只能包含字母、数字、下划线、"-"

操作权限	用户	IP或网段	策略
允许 ▾	请选择, 不选默 ▾	请输入IP或网段, 支持";"分隔, 且	写 ▾
<a href="#" style="color: #007bff; text-decoration: none;">添加规则</a>			

自动应用到后续所有新增topic

提交
关闭

后续处理：完成授权后，用户可以通过 SASL 接入点接入消息队列 CKafka 并使用 PLAIN 机制消费消息（参见 [SDK 文档](#)）。

## 使用限制

1. 开通路由只影响接入时的验证方式，设置的 ACL 权限则是全局的。
2. 如果您在开通公网访问路由的同时还使用了 PLAINTEXT 方式接入 CKafka，那么之前为 Topic 设置的 ACL 仍然会生效。若您希望 PLAINTEXT 方式的访问不受影响，请为 PLAINTEXT 需要访问的 Topic 添加全部用户的可读写的权限。

### 说明

在添加 ACL策略时，不需要选择任何用户，默认为全部用户添加了读写权限。

操作权限	用户	IP或网段	策略
允许 ▾	请选择, 不选默认全部 ▾	请输入IP或网段, 支持";"分隔, 默认*	写 ▾
<a href="#" style="color: #007bff; text-decoration: none;">添加规则</a>			

添加完成效果如下：

权限类型	用户	IP	策略	资源名	操作
允许	全部	全部	写	clue-test	<a href="#" style="color: #007bff; text-decoration: none;">删除</a>
允许	全部	全部	读	clue-test	<a href="#" style="color: #007bff; text-decoration: none;">删除</a>

3. 如果该 Topic 已经有其他云产品在使用（例如：日志服务 CLS 的日志投递、云函数 SCF 消息转储、大数据 EMR 组件的消费等），开启 ACL 策略相当于对这些联动能力的权限加以限制，会直接导致这些能力不可用，请一定谨慎操作。对于此类情况建议生产同一份数据到另一个 Topic 做分别处理，不要在同一个 Topic 上配置统一的 ACL 策略。

## 查看预设规则

1. 在 ACL 策略管理页面，选择**预设规则**。
2. 在预设规则列表，单击操作列的**详情**，可查看预设规则详情。

### 规则详情

规则名称 test1

应用topic [topic\\_acl等4个](#)

ACL策略 允许 User:\* 通过 \* 写 上述Topic

自动应用至后续所有新增topic

## 删除预设规则

1. 在 ACL 策略管理页面，选择**预设规则**。
2. 在预设规则列表，单击操作列的**删除**，可删除预设规则。

根据规则匹配类型不同，预设规则删除后的影响也有所不同：

策略列表		用户管理	
批量配置	资源	预设规则	
规则名称	匹配类型	操作	
test1	预设策略	<a href="#">删除</a> <a href="#">详情</a>	
test2	前缀匹配 前缀: *	<a href="#">删除</a> <a href="#">详情</a>	
test3	预设策略	<a href="#">删除</a> <a href="#">详情</a>	

- 当规则是模糊匹配规则时，新增 Topic 不会再自动应用该规则，对于已经应用前缀匹配规则的 Topic，该规则也将不再生效。
- 当规则不是模糊匹配规则时，新增 Topic 不会再自动应用该规则，对于已自动应用该规则的 Topic，相应规则不会一并删除。

# 访问管理 CAM

Last updated: 2023-04-11 09:54:29



## CAM 基本概念

主账号通过给予子账号绑定策略实现授权，策略设置可精确到 [API, 资源, 用户/用户组, 允许/拒绝, 条件] 维度。

### 账户

- **主账号**：拥有腾讯云所有资源，可以任意访问其任何资源。
- **子账号**：包括子用户和协作者。
- **子用户**：由主账号创建，完全归属于创建该子用户的主账号。
- **协作者**：本身拥有主账号身份，被添加作为当前主账号的协作者，则为当前主账号的子账号之一，可切换回主账号身份。
- **身份凭证**：包括登录凭证和访问证书两种，**登录凭证**指用户登录名和密码，**访问证书**指云 API 密钥（SecretId 和 SecretKey）。

### 资源与权限

- **资源**：资源是云服务中被操作的对象，如一个云服务器实例、COS 存储桶、VPC 实例等。
- **权限**：权限是指允许或拒绝某些用户执行某些操作。默认情况下，**主账号拥有其名下所有资源的访问权限，而子账号没有主账号下任何资源的访问权限。**
- **策略**：策略是定义和描述一条或多条权限的语法规则。**主账号通过将策略关联到用户/用户组完成授权。**

[单击查看更多 CAM 文档>>](#)

## 相关文档

目标	链接
了解策略和用户之间关系	<a href="#">策略管理</a>
了解策略的基本结构	<a href="#">策略语法</a>
了解还有哪些产品支持 CAM	<a href="#">支持 CAM 的产品</a>

## 访问控制策略示例

### CKafka 全读写策略

授权一个子用户以 CKafka 服务的完全管理权限（创建、管理等全部操作）。

```
{
  "version": "2.0",
  "statement": [
    {
      "action": [
        "name/ckafka:*",
        "name/monitor:GetMonitorData"
      ],
      "resource": "*",
      "effect": "allow"
    }
  ]
}
```

您也可以通过设置系统的 [全读写策略](#) 支持。

1. 登录 [访问管理控制台](#)
2. 在左侧菜单栏中，单击 [策略](#)。
3. 在策略列表中，单击 [新建自定义策略](#)。
4. 在选择创建策略方式的弹窗中，选择 [按策略语法创建](#)。
5. 在模板类型中，搜索“CKafka”，选择消息服务（CKafka）全读写访问权限 [QcloudCKafkaFullAccess]，单击 [下一步](#)。
6. 单击 [完成](#)。

## CKafka 实例只读策略

1. 按照策略生成器创建，授权列表类权限和产品监控权限。

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "action": [
        "name/ckafka:ListInstance",
        "name/monitor:GetMonitorData"
      ],
      "resource": [
        "*"
      ]
    }
  ]
}
```

2. 授权单实例只读权限

### 注意

List\* 接口不支持资源粒度的鉴权。

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "action": [
        "name/monitor:GetMonitorData",
        "name/ckafka:Get*"
      ],
      "resource": [
        "qcs::ckafka:gz::ckafkald/uin/$createUin/$instanceId"
      ]
    }
  ]
}
```

您也可以通过设置系统的 [只读策略](#) 支持。

1. 登录 [访问管理控制台](#)
2. 在左侧菜单栏中，单击 [策略](#)。
3. 在策略列表中，单击 [新建自定义策略](#)。
4. 在选择创建策略方式的弹窗中，选择 [按策略语法创建](#)。
5. 在模板类型中，搜索“CKafka”，选择消息服务（CKafka）只读访问策略 [QcloudCkafkaReadOnlyAccess]，单击 [下一步](#)。
6. 单击 [完成](#)。

# 标签管理

## 标签概述

Last updated: 2021-12-28 10:09:35

### 简介

标签是腾讯云提供的用于标识云上资源的标记，是一个键-值对（Key-Value）。

您可以根据各种维度（例如业务、用途、负责人等）使用标签对 CKafka 消息队列资源进行分类管理，通过标签非常方便地筛选过滤出对应的资源。标签键值对对腾讯云没有任何语义意义，会严格按字符串进行解析匹配。

### 使用限制

#### 数量限制

每个云资源允许的最大标签数是50。

#### 标签键限制

- qcloud、tencent、project 开头为系统预留标签键，禁止创建。
- 只能为字母、数字、空格或汉字，支持 +、-、=、.、\_、:、/、@。
- 标签键长度最大为255个字符。

#### 标签值限制

- 只能为字母、数字、空格或汉字，支持 +、-、=、.、\_、:、/、@。
- 标签值最大长度为127个字符。

### 使用示例

#### 案例背景

某公司在腾讯云上拥有10个 CKafka 消息队列实例，分属电商、游戏、文娱三个部门，服务于营销活动、游戏 A、游戏 B、后期制作等业务，三个部门对应的运维负责人为张三、李四、王五。

#### 设置标签

为了方便管理，该公司使用标签分类管理对应的 CKafka 消息队列资源，定义了下述标签键/值。

标签键	标签值
部门	电商、游戏、文娱
业务	营销活动、游戏 A、游戏 B、后期制作
运维负责人	张三、李四、王五

将这些标签键/值绑定到 CKafka 上，资源与标签键/值的关系如下表所示：

ID	部门	业务	运维负责人
ckafka-1jqwv1	电商	营销活动	王五
ckafka-1jqwv12	电商	营销活动	王五
ckafka-1jqwv13	游戏	游戏 A	张三
ckafka-1jqwv13	游戏	游戏 B	张三
ckafka-1jqwv14	游戏	游戏 B	张三
ckafka-1jqwv15	游戏	游戏 B	李四
ckafka-1jqwv16	游戏	游戏 B	李四

ckafka-1jqwv17	游戏	游戏 B	李四
ckafka-1jqwv18	文娱	后期制作	王五
ckafka-1jqwv19	文娱	后期制作	王五
ckafka-1jqwv110	文娱	后期制作	王五

## 使用标签

- 筛选出王五负责的 CKafka 实例：  
按照筛选规则筛选出运维负责人为“王五”的 CKafka 资源即可，具体筛选步骤请参见 [使用标签](#)。
- 筛选出游戏部门中李四负责的 CKafka 实例：  
按照筛选规则筛选出部门为“游戏”、运维负责人为“李四”的 CKafka 资源即可，具体筛选步骤请参见 [使用标签](#)。

## 使用标签

Last updated: 2021-08-25 14:25:44

### 操作场景

该任务指导您在消息队列 CKafka 控制台中，根据标签对实例进行资源筛选，过滤出对应的资源。

### 操作步骤

1. 登录 [消息队列 CKafka](#) 控制台。
2. 在实例列表页顶部，选择地域。
3. 在实例列表右上角的搜索框，单击空白处，选择**标签**。
4. 在“标签：”后输入标签键。  
例如：筛选出绑定了某个标签键（如 name）的实例，输入“name”（区分大小写）。
5. 回车或单击搜索 icon，进行搜索。

## 编辑标签

Last updated: 2021-08-25 14:26:58

### 操作场景

该任务指导您在消息队列 CKafka 控制台中，对目标实例进行标签的编辑操作。

### 操作步骤

#### ① 说明

您可以对单个或多个实例编辑标签，最多支持对20个实例进行标签的批量编辑操作。

1. 登录 [消息队列 CKafka](#) 控制台。
2. 在 CKafka 实例列表页，勾选需要编辑标签的实例，单击列表上方的**编辑标签**。
3. 在编辑标签的操作框中，添加、修改或删除标签。
4. 单击**确定**，完成编辑。

# 查询消息

Last updated: 2023-04-11 09:54:29

## 操作场景

若您遇到消息消费异常，可以在 CKafka 控制台查询异常消息来排查问题。该任务指导您通过 CKafka 控制台查询消息。

CKafka 控制台支持按位点查询和按时间查询两种方式查询消息，适用场景如下：

- 按位点查询：用户能明确知道消息发送至 Topic 的分区 ID 以及消息位点。
- 按时间查询：用户不确定消息的位置，但确定消息发送的时间。

### 注意

- 单次查询最多查询起始时间或者起始位点之后的20条消息，并不会列出所有消息。
- 查询消息也会消耗一定的带宽资源，切勿频繁操作。

## 操作步骤

### 按位点查询

- 登录 [CKafka 控制台](#)。
- 在左侧导航栏选择**实例列表**，选择地域后，单击目标实例的“ID/名称”，进入实例详情页。
- 在实例详情页，选择 **topic 管理**，单击目标 Topic 右侧操作列的**更多 > 消息查询**。
- 在消息查询页面，选择**按位点查询**，选择需要查询的分区 ID 和起始位点，单击**查询**，查看消息信息。

消息查询会占用CKafka实例的带宽资源，建议您尽量缩小查询范围，不要频繁操作。  
消息查询最多展示指定位点或时间后的20条数据。

实例: ckafka-7c...  
Topic: test

查询类型: **按位点查询** | 按起始时间查询

分区ID: 0  
起始位点: 1

**查询**

分区ID	位点	时间戳	操作
0	1	2022-08-08 19:54:20	<a href="#">查看详情</a> <a href="#">下载消息</a>

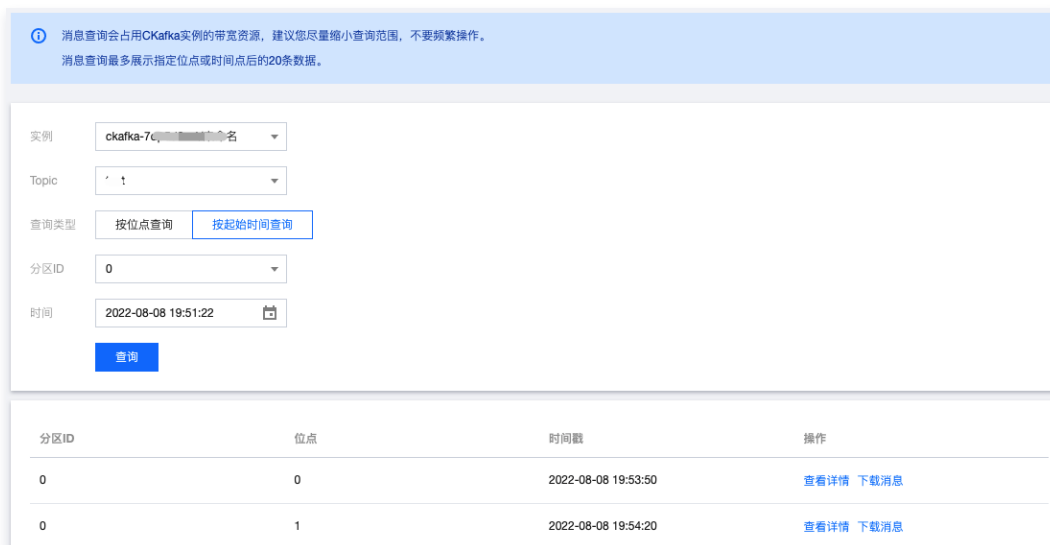
- 分区 ID：消息的 Topic 分区。
- 位点：消息的消费位点。
- 时间戳：为 `ProducerRecord` 中的 `timestamp`。
- 操作：
  - 单击**下载消息**可以下载消息，格式为 header, key, value。
  - 单击**消息详情**可查看消息的 key 和 value。



## 按时间查询

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例列表**，选择地域后，单击目标实例的“ID/名称”，进入实例详情页。
3. 在实例详情页，选择 **topic 管理**，单击目标 Topic 右侧操作列的**更多 > 消息查询**。
4. 在消息查询页面，选择**按起始时间查询**，选择需要查询的分区 ID 和时间，单击**查询**，查看消息信息。

如果选择全部分区，则显示全部分区的消息；如果指定具体分区，则显示指定分区的消息。



- 分区 ID：消息的 Topic 分区。
- 位点：消息的消费位点。
- 时间戳：为 `ProducerRecord` 中的 `timestamp`。
- 操作：
  - 单击**下载消息**可以下载消息，包含消息的 Headers, Key, Value。
  - 单击**查看详情**可查看消息详情。

消息详情 ✕

**i** 当前查询的消息已经被强制转换为String类型，如出现乱码，请分析您消息的序列化格式以及编码格式

Headers	暂无数据
Key	hello
Value	hello world

确定

# 事件中心

Last updated: 2023-05-24 16:59:10

事件中心将 CKafka 所生成的事件数据进行统一管理、存储、分析和展示，方便您查看和分析，您可以在事件中心查看事件数据详情。

CKafka 当前支持的事件类型有：

- 实例事件
  - 磁盘自动扩容
  - 磁盘动态消息保留策略
  - 分区均衡
  - Kafka 版本升级
  - 内核小版本升级
  - 可用区变更
- 诊断事件
  - 定时实例诊断
  - 即时实例诊断
- Broker 变更
  - 消息分组成员心跳超时
  - 消息分组成员更新
  - 消息分组 rebalance
  - Leader 切换
  - 集群节点上线
  - 集群节点下线

本文介绍如何在 CKafka 控制台上查看事件详情和配置事件通知规则。

## 进入事件中心

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择**实例管理**，单击目标实例的“ID”，进入实例详情页面。
3. 在页面上方选择“事件中心”页签，设置好时间范围（支持近6小时、近7天、近30天和自定义时间范围），选择需要查看的事件类型。



4. 在事件详情页面，您可以进行以下操作：

- 单击操作列的**查看详情**，您可以在右侧栏查看事件详情。



- 单击操作列的**配置告警**，您可以在事件总线为事件配置通知模板，将事件详情发送给目标对象。

# 迁移上云

## 迁移方案概述

Last updated: 2021-08-26 15:08:18

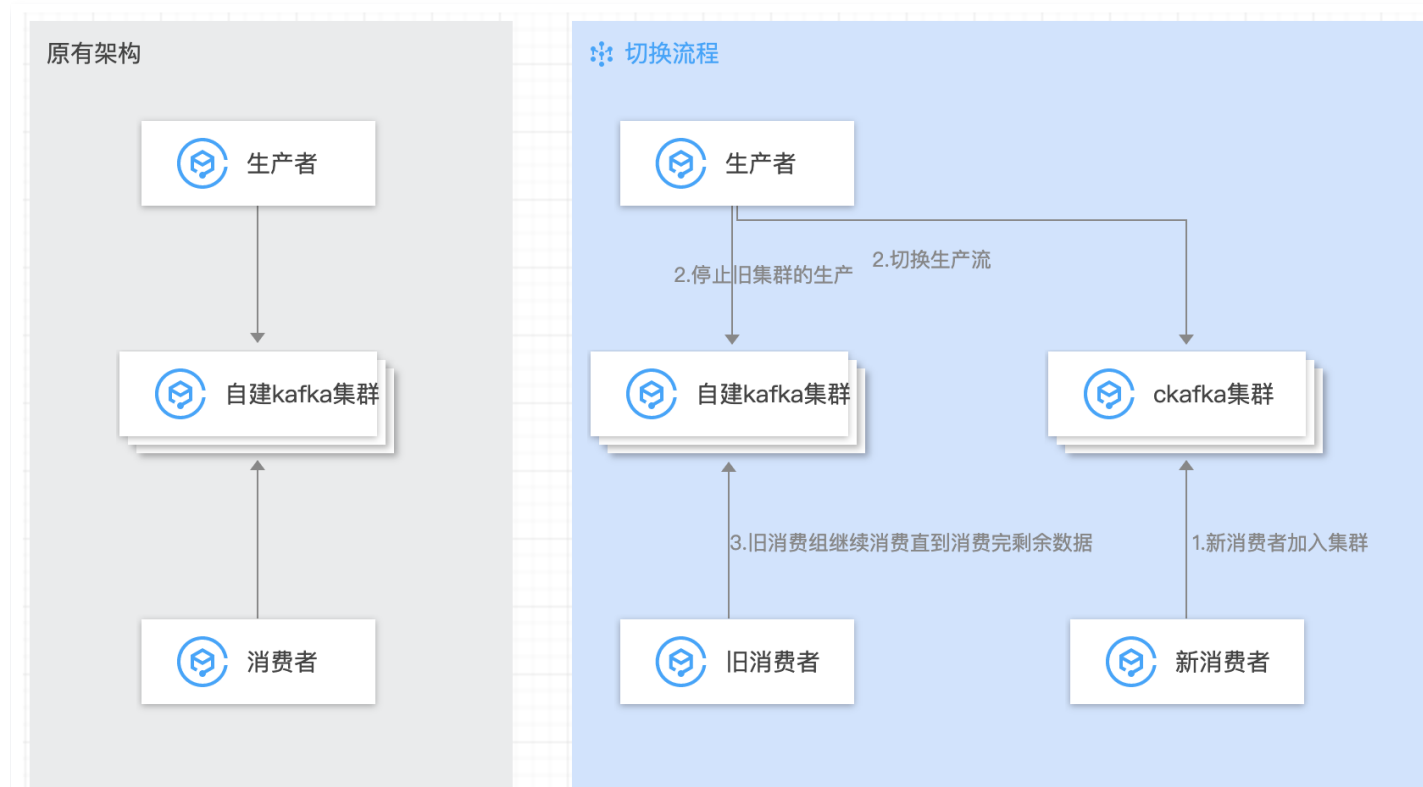
### 操作场景

本文档为您介绍将自建 Kafka 集群迁移到 CKafka 集群的可行方案，您可以根据自身业务场景选择合适您的迁移方案。

### 方案说明

#### 方案1：单写双消费

该方案的整体简单清晰便于操作，且无数据积压，过渡平滑。



#### 方案思路：

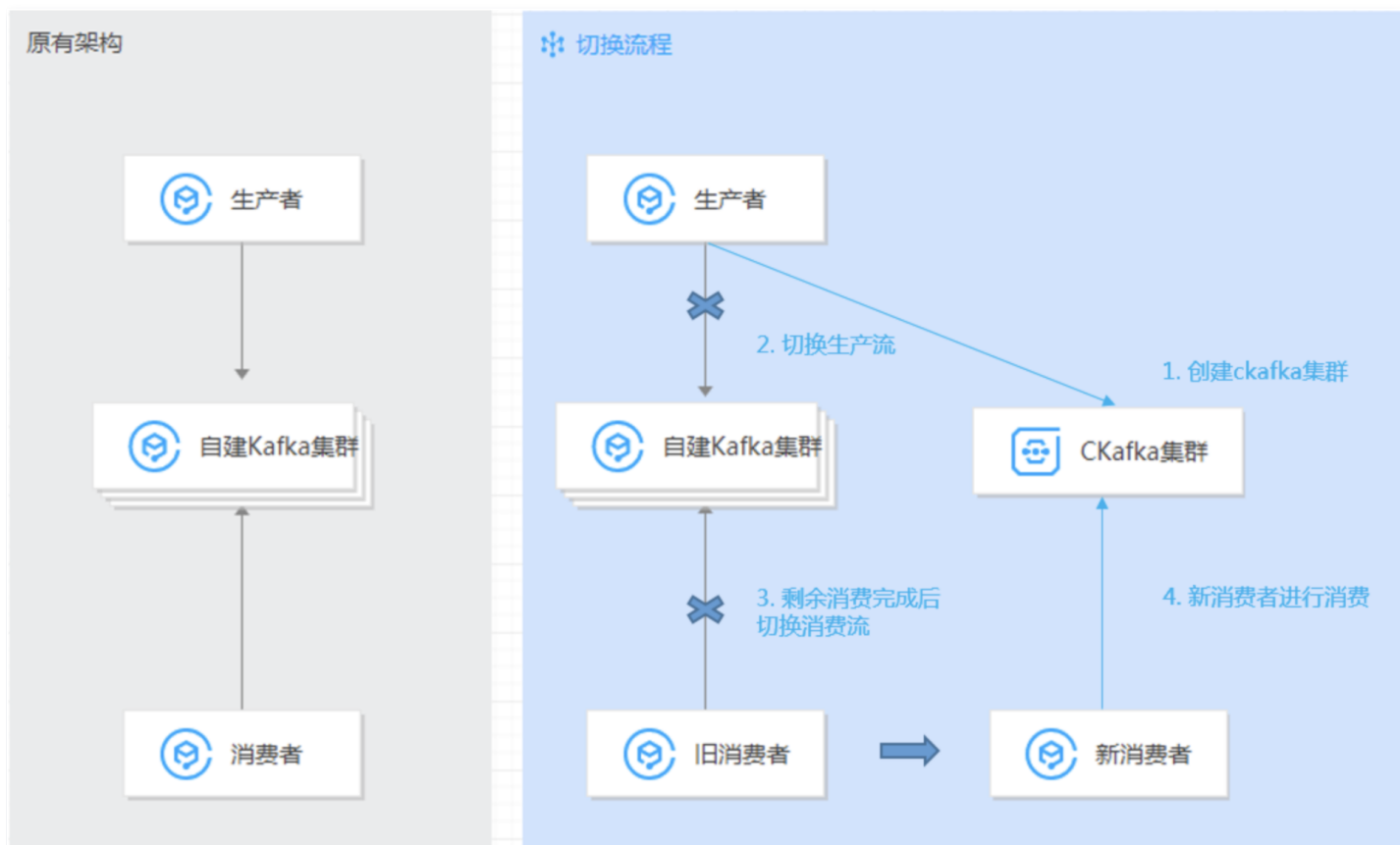
1. 完成 Topic 元数据的迁移。
2. 自建 Kafka 集群中原有的消费者保持不动。
3. CKafka 消费端新起消费者，配置新的 CKafka 集群的 bootstrap-server，消费新的 CKafka 集群。
4. 等待所有消费端都已经监听了新的 CKafka 集群。
5. 将自建集群的生产切到 CKafka 新集群上（配置新的 CKafka 集群的 bootstrap-server）。
6. 自建 Kafka 集群中原有的消费者继续消费自建 Kafka 集群中剩余的数据，直到消费干净后方可下线原消费者。

#### 方案优劣：

- 优点：整体迁移流程简单清晰便于操作，无数据积压，平滑过渡。
- 缺点：需要额外多起一套消费者。

#### 方案2：单写单消费

该方案的整体简单清晰便于操作。

**方案思路:**

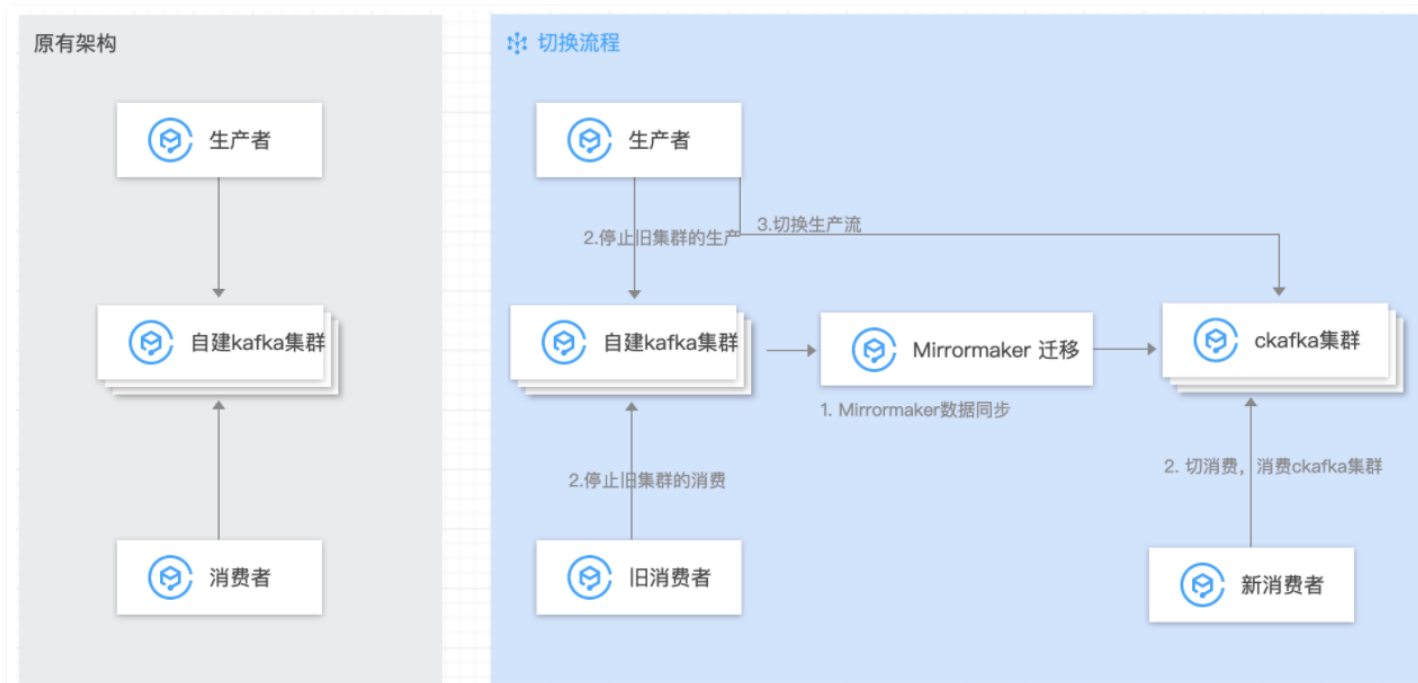
1. 完成 Topic 元数据的迁移。
2. 将自建 Kafka 集群的生产切到 CKafka 新集群上 (配置新的 CKafka 集群的 bootstrap-server)。
3. 等待自建集群中的消费者消费完剩余数据。
4. 将老的消费者切到 CKafka 新集群消费 (配置新的 CKafka 集群的 bootstrap-server)。

**方案优劣:**

- 优点: 整体迁移流程简单清晰便于操作, 过渡平滑。
- 缺点: 在生产切到 CKafka 集群后, 旧消费切到 CKafka 集群之前, CKafka 集群会存在一定量的堆积。

**方案3: Mirrormaker 迁移**

该方案会把自建集群 Kafka 中的存量数据迁移到 CKafka。

**方案思路：**

1. 完成 Topic 元数据的迁移。
2. 自建 Kafka 集群中原有的消费者保持不动。
3. 启动 [Mirrormaker 工具](#) 的数据同步功能。
4. 等待数据同步完成，修改消费者配置并切换消费者。
5. 等待数据同步完成，修改生产者配置并切换生产者。
6. 迁移完成。

**方案优劣：**

- 优点：整体迁移流程简单清晰便于操作、可以把历史数据同步到 CKafka 集群。
- 缺点：消费者切换到目的集群上需要从头开始消费，需要做好消费幂等。

## 步骤1：购买云上实例

Last updated: 2023-05-04 15:00:20

### 操作场景

本文档为您介绍如何利用 CKafka 控制台上的规格计算器为您的自建 Kafka 集群迁移上云评估合适的云上实例规格。

### 操作步骤

1. 登录 [CKafka控制台](#)。
2. 在左侧导航栏选择**迁移上云**，选择您要迁移的地域后单击**规格计算器**。
3. 在规格计算器页面输入您的自建Kafka集群规格。

规格计算器
×

1 输入自建规格
 >
2 推荐规格

Kafka版本 0.10.x [版本选择建议](#)

0.10及以下的版本选择0.10.x

业务带宽峰值 40 MB/s

业务带宽峰值 = max(生产带宽峰值\*副本数, 消费带宽峰值)

磁盘 300 GB

按照当前实际磁盘堆积峰值评估

分区总数 60 个

需要迁移的topic的分区总数。注意需要考虑副本数。例如一个topic单副本时是5个分区，则双副本情况下，分区总数是10。CKafka不支持单副本的topic

跨可用区部署  是  否

数据压缩 未开启

CKafka不支持Gzip压缩格式，详细内容参考 [数据压缩](#)

下一步

参数	说明
Kafka 版本	选择您的自建 Kafka 版本，CKafka 版本选择可参见 <a href="#">版本选择建议</a> 。
业务带宽峰值	业务带宽峰值 = max ( 生产带宽峰值 × 副本数, 消费带宽峰值 )。
磁盘	按照当前实际磁盘堆积峰值评估。
分区总数	需要迁移的 Topic 的分区总数。注意需要考虑副本数，例如一个 Topic 单副本时是5个分区，则双副本情况下，分区总数是10。CKafka 不支持单副本的 Topic。
跨可用区部署	根据您的业务需求选择是否要进行跨可用区部署，详情请参见 <a href="#">跨可用区部署</a> 。
数据压缩	CKafka 不支持 Gzip 压缩格式，详情请参见 <a href="#">数据压缩</a> 。

4. 单击**下一步**获取为您推荐 CKafka 实例规格。
5. 单击**购买此配置**，跳转至实例购买页面。
6. 确认购买信息后，单击**立即购买**，等待5-10分钟后，在实例列表页可以看到实例创建完成。

<input type="button" value="新建"/> <input type="button" value="编辑标签"/> <input type="button" value="删除"/> <input type="button" value="销毁"/> <input type="text" value="请输入关键字进行搜索"/>									
ID/名称	监控	状态	可用区	实例类型	配置	网络类型	计费模式	标签	操作
<input type="checkbox"/> ckafka- 未命名		健康	上海三区	专业版-基础型	Topic数量上限: 50个 Partition数量上限: 500个 版本: 0.10.2 磁盘类型: 高性能云硬盘	私有网络	包年包月 2021-02-08到期		<a href="#">配置告警</a> <a href="#">续费</a> <a href="#">更多</a>

## 步骤2：迁移 Topic 上云

Last updated: 2023-06-06 17:10:01

### 操作场景

本文档为您介绍如何利用 CKafka 提供的迁移工具将自建 Kafka 集群的 Topic 迁移到 CKafka 的实例中。

### 前提条件

- 已 [购买云上实例](#)
- 下载 [Python2](#)

### 操作步骤

- 下载 [迁移工具](#) 并解压到可以连通自建实例的 broker 和 zk 的机器上。
- 在 ckafka-migrate.py 文件中填写配置参数。

#### 说明

- 请保证后面迁移操作所在的机器和 CKafka 以及自建 Kafka 集群的网络互通。
- 云API的密钥对应的用户需要拥有 CKafka 的写权限，建议使用主账号的密钥对。

```
# your local broker ip:port list
# 自建实例的broker列表 ["broker1:port1","ip2:port2"]
bootstrapServers = ["$ip:$port"]

# your local zk ip:port list
# 自建实例的zk列表 ["zk1:port1","zk2:port2"],非必设配置,未设置则用bootstrapServers方式获取源集群信息
sourceZk = ["$ip:$port"]
# 如果zk有auth鉴权,需要设置该参数,格式为 [("digest", "$user:$password")]
zkAuthData = []
# 如果zk路径有前缀,需要设置该参数,例如 "/cluster1"
zkPathPrefix = ""

# your cloud instanceId
# 云上实例id "ckafka-xxx"
instanceId = "$yourinstanceId"
# topic regex,just migrate match topics
# topic名称正则表达式,非空则只迁移匹配到的topic
topicRegex = ""

# your secretId and secretKey
# 账号的密钥对
secretId = "$yoursecretId"
secretKey = "$yoursecretKey"

# your cloud instance region
# 云上实例的地域 ckafka已开区域地域码:
# 广州 ap-guangzhou;上海 ap-shanghai;南京 ap-nanjing;北京 ap-beijing; 成都 ap-chengdu;重庆 ap-chongqing;
# 香港 ap-hongkong;新加坡 ap-singapore;印度孟买 ap-mumbai;日本东京 ap-tokyo;美西硅谷 na-siliconvalley;
# 美东弗吉尼 na-ashburn;北美多伦多 na-toronto;中国台北 ap-taipei;天津 ap-tianjin;上海金融 ap-shanghai-fsi;
# 深圳金融 ap-shenzhen-fsi;深圳 ap-shenzhen;德国法兰克福 eu-frankfur;首尔 ap-seoul;清远 ap-qingyuan;
# 北京金融 ap-beijing-fsi;曼谷 ap-bangkok;长沙 ap-changsha-ec;雅加达 ap-jakarta
# 杭州ec ap-hangzhou-ec;圣保罗 sa-saopaulo;上海自动驾驶 ap-shanghai-adc;武汉ec ap-wuhan-ec;西安ec ap-xian-ec;
region = "ap-tokyo"

# if you make sure the migrate topic List,please modify checkFlag = 1
# 检查标记,设0只显示将要迁移的topic列表不做真正迁移,请先以0运行检查将要迁移的topic列表, 确认无误后修改为1开始迁移
# 0:列出迁移topic列表后脚本终止
# 1:列出迁移topic列表并开始迁移
checkFlag = 0
```

```
# force transfor your cloud-topic config to migrate
# 如果为0转换本地topic到云上topic时, 属性不一致不会迁移上云。如果为1, 会强制转换topic属性和云上最近的值
# 例如云上topic的副本只支持1, 2, 3副本, 如果本地的某topic副本数为5, 则不会迁移上云。如果force设置为1, 则会取和5最接近的3副本, 在
# 0:本地和云上topic副本数或topic属性不兼容时, 跳过不兼容的topic或者topic属性
# 1:本地和云上topic副本数或topic属性不兼容时, 强制迁移至云上, 云上topic的属性将根据自建topic的属性做出最接近的修正 (不会修改本地)
force = 0
```

参数	说明
bootstrapServers	自建实例的broker列表, ["ip1:port1","ip2:port2"]。
sourceZk	自建实例的zookeeper列表, ["zk1:port1","zk2:port2"], 非必配置,未设置则用bootstrapServers方式获取源集群信息
zkAuthData	如果zk有auth鉴权, 需要设置该参数,格式为 [("digest", "\$user:\$password")]。
zkPathPrefix	如果zk路径有前缀, 需要设置该参数,例如 "/cluster1"。
instanceId	您在 <a href="#">购买云上实例</a> 中购买的 CKafka 实例的 ID, 在控制台的 <a href="#">实例列表</a> 页面复制。
secretId	账号的密钥对-ID。
secretKey	账号的密钥对-密码。
region	您在 <a href="#">购买云上实例</a> 中选择的部署地域, 脚本内注释附带各地域码。
checkFlag	检查标记, 设置为0时只显示将要迁移的 Topic 列表并不开始迁移, 设置为非0时开始迁移 Topic。
topicRegex	Topic 名称正则表达式, 设置为空时迁移所有的 Topic, 非空时则只迁移匹配到的 Topic。
force	是否强制迁移, 如果为0转换本地 Topic 到云上 Topic 时, 属性不一致不会迁移上云。如果为1, 会强制转换 Topic 属性和云上规定最接近的值。

3. 将 kafka-migrate.py 的 checkFlag 参数设为0, 运行脚本 python kafka-migrate.py, 根据输出结果检查需要迁移的Topic列表。

① 说明

如果缺少部分自建的 Topic, 可能是自建Topic命名不符合规则或者Topic副本数、Topic属性数值与云上数值范围无法兼容。

```
[root@M-0-21-centos ~/migratecloud/migrateToCkafkaTool]# python kafka-migrate.py
check topic list to migrate: ["test1", "test0", "test108", "test109", "test5", "test4", "test7", "test6", "test102", "test103", "test100",
"test101", "test106", "test107", "test104", "test105", "test182", "test183", "test180", "test181", "test186", "test187", "test184", "te
st185", "test188", "test189", "test46", "test47", "test44", "test45", "test42", "test43", "test40", "test41", "test48", "test49", "test8", "
test179", "test178", "test177", "test176", "test175", "test174", "test173", "test172", "test171", "test170", "test55", "test54", "test57",
"test56", "test51", "test50", "test53", "test52", "test59", "test58", "test168", "test169", "test164", "test165", "test166", "test167", "
test160", "test161", "test162", "test163", "test160", "test161", "test162", "test163", "test164", "test165", "test166", "test167", "test
69", "test2", "test159", "test158", "test151", "test150", "test153", "test152", "test155", "test154", "test157", "test156", "test99", "tes
t78", "test77", "test76", "test75", "test74", "test73", "test72", "test71", "test70", "test146", "test147", "test144", "test145", "test142
", "test143", "test140", "test141", "test148", "test149", "test82", "test83", "test80", "test81", "test86", "test87", "test84", "test85", "
test88", "test89", "test3", "test133", "test132", "test131", "test130", "test137", "test136", "test135", "test134", "test139", "test138",
"test91", "test90", "test93", "test92", "test95", "test94", "test97", "test96", "test99", "test98", "test35", "test19", "test18", "test11
", "test10", "test11", "test12", "test15", "test14", "test17", "test16", "test120", "test121", "test122", "test123", "test124", "test125",
"test126", "test127", "test128", "test129", "test24", "test25", "test26", "test27", "test20", "test21", "test22", "test23", "test28", "te
st29", "test", "test118", "test119", "test115", "test114", "test117", "test116", "test111", "test110", "test113", "test112", "test195", "te
st194", "test197", "test196", "test191", "test190", "test193", "test192", "test199", "test198", "test33", "test32", "test31", "test30", "
test37", "test36", "test9", "test34", "test39", "test38"]
```

4. 把 kafka-migrate.py 的 checkFlag 参数设为1, 运行脚本 python kafka-migrate.py, 开始迁移 Topic。

```
[root@M-0-21-centos ~/migratecloud/migrateToCkafkaTool]# python kafka-migrate.py
preper to migrate these topics ["test1", "test0", "test108", "test109", "test5", "test4", "test7", "test6", "test102", "test103", "test100
", "test101", "test106", "test107", "test104", "test105", "test182", "test183", "test180", "test181", "test186", "test187", "test184", "te
st185", "test188", "test189", "test46", "test47", "test44", "test45", "test42", "test43", "test40", "test41", "test48", "test49", "test8", "
test179", "test178", "test177", "test176", "test175", "test174", "test173", "test172", "test171", "test170", "test55", "test54", "test57
", "test56", "test51", "test50", "test53", "test52", "test59", "test58", "test168", "test169", "test164", "test165", "test166", "test167", "te
st160", "test161", "test162", "test163", "test160", "test161", "test162", "test163", "test164", "test165", "test166", "test167", "test
69", "test2", "test159", "test158", "test151", "test150", "test153", "test152", "test155", "test154", "test157", "test156", "test179", "te
st78", "test77", "test76", "test75", "test74", "test73", "test72", "test71", "test70", "test146", "test147", "test144", "test145", "test142
", "test143", "test140", "test141", "test148", "test149", "test82", "test83", "test80", "test81", "test86", "test87", "test84", "test85", "
test88", "test89", "test3", "test133", "test132", "test131", "test130", "test137", "test136", "test135", "test134", "test139", "test138
", "test91", "test90", "test93", "test92", "test95", "test94", "test97", "test96", "test99", "test98", "test35", "test19", "test18", "test11
", "test10", "test11", "test12", "test15", "test14", "test17", "test16", "test120", "test121", "test122", "test123", "test124", "test125
", "test126", "test127", "test128", "test129", "test24", "test25", "test26", "test27", "test20", "test21", "test22", "test23", "test28", "te
st29", "test1", "test118", "test119", "test115", "test114", "test117", "test116", "test111", "test110", "test113", "test112", "test195", "te
st194", "test197", "test196", "test191", "test190", "test193", "test192", "test199", "test198", "test33", "test32", "test31", "test30", "
test37", "test36", "test9", "test34", "test39", "test38"] to cloud instance kafka-lxmap3v4 .....
create Migrate to Cloud Task Success!
1/201 migrate topic test1 to kafka-lxmap3v4 success,topicId:topic-9c9q3121
2/201 migrate topic test0 to kafka-lxmap3v4 success,topicId:topic-057vfbw2
```

5. 登录 CKafka 控制台, 在迁移上云页面查看任务列表, 等待 Topic 迁移完毕。

任务列表如下:

自建上云迁移 长沙(4)

规格计算器 实例 ID/名称

ID/名称	可用区	Topic迁移	Consumer Group迁移	数据迁移
ckafka- 自建上云Demo	450001	201 / 201	0 / 176	无

迁移成功界面如下:

```
199/201 migrate topic test34 to ckafka-lxmap3v4 success,topicId:topic-a4bo0ske
200/201 migrate topic test39 to ckafka-lxmap3v4 success,topicId:topic-bv7pezvi
201/201 migrate topic test38 to ckafka-lxmap3v4 success,topicId:topic-m96t1b0e
migrate topics to cloud finished!
```

## 步骤3：迁移数据上云

### 方案1：单写双消费迁移

Last updated: 2021-08-26 15:07:59

#### 操作场景

本文主要介绍使用单写双消费方案将自建 Kafka 集群的数据迁移到 CKafka 中的方法。

#### 前提条件

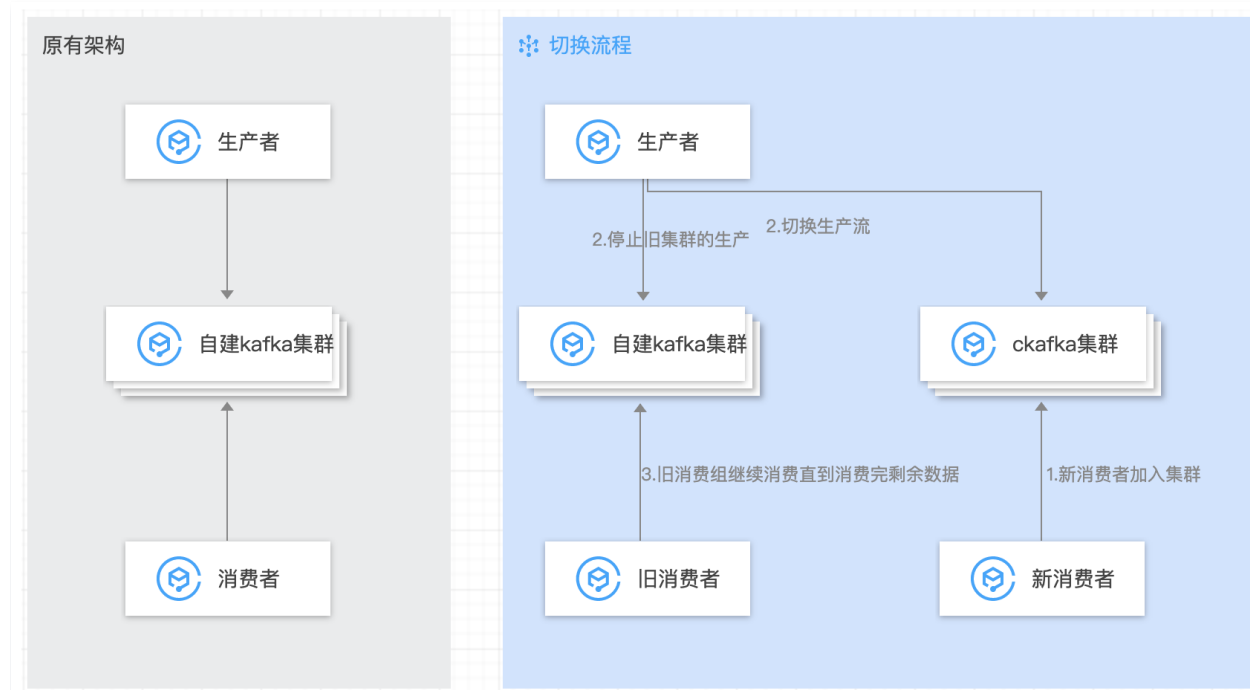
- 已 [购买云上 CKafka 实例](#)。
- 已 [迁移 Topic 上云](#)。

#### 操作步骤

对于数据有序性要求不高的情况下，可以采用多个消费者并行消费的方式进行切换。

单写双消费的方式简单清晰便于操作且无数据积压，平滑过渡；但是需要业务侧新增一套消费者。

其迁移步骤如下所示：



1. 旧的消费者保持不动，消费端新起消费者，配置新的集群的 bootstrap-server，消费新的 CKafka 集群。

需要配置 `--bootstrap-server` 中的 IP 为 CKafka 实例的接入网络，在控制台的实例详情页面接入方式模块的网络列复制。

```
./kafka-console-consumer.sh --bootstrap-server xxx.xxx.xxx.xxx:9092 --from-beginning --new-consumer --topic topicName --consi
```

2. 切换生产流，生产者将数据生产到 CKafka 实例。

修改 `broker-list` 中的 IP 为 CKafka 实例的接入网络，topicName 为 CKafka 实例中的 Topic 名称：

```
./kafka-console-producer.sh --broker-list xxx.xxx.xxx.xxx:9092 --topic topicName
```

3. 原有消费者无需特殊配置，继续消费自建 Kafka 集群的数据。当原有自建集群的数据消费完成后，即迁移完毕。

#### 注意

上文给出的是测试命令，正式业务的运行只需要修改相应应用程序配置的 broker 地址，然后重启相应的应用即可。

## 方案2：单写单消费迁移

Last updated: 2021-09-01 16:55:08

### 操作场景

本文主要介绍使用单写单消费方案将自建 Kafka 集群的数据迁移到 CKafka 中的方法。

### 前提条件

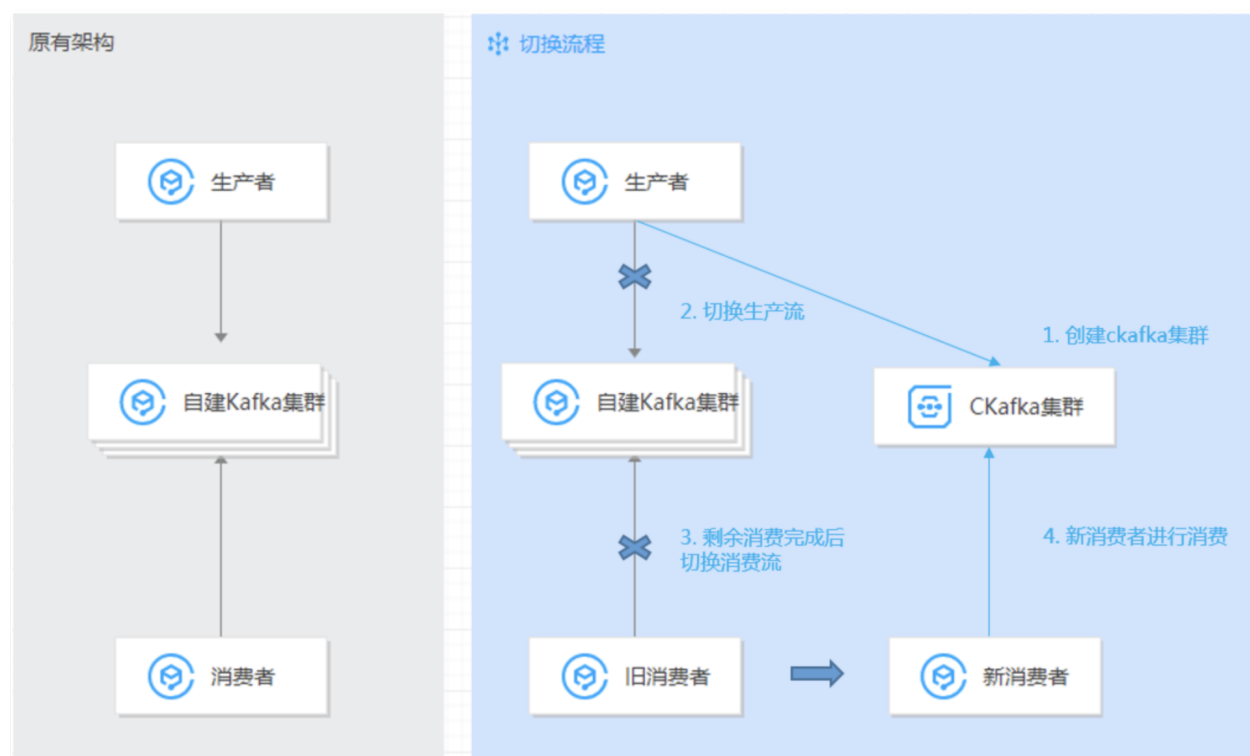
- 已 [购买云上 CKafka 实例](#)。
- 已 [迁移 Topic 上云](#)。

### 操作步骤

保证消息有序性的前提是严格控制单个消费者来消费数据，因此对于切换的时间节点要求较高。

单写单消费的方式简单清晰便于操作，但是在生产切到新集群后，旧消费切到新集群之前，新集群会存在一定量的堆积。

其迁移步骤如下所示：



1. 切换生产流，生产者将数据生产到 CKafka 实例。

修改 `broker-list` 中的 IP 为 CKafka 实例的接入网络，在控制台的实例详情页面[接入方式](#)模块的网络列复制；`topicName` 为 CKafka 实例中的 Topic 名称。

```
./kafka-console-producer.sh --broker-list xxx.xxx.xxx.xxx:9092 --topic topicName
```

2. 原有消费者无需做配置，持续消费自建 Kafka 集群的数据，直到消费完成。

3. 原有消费者在消费完成时，通过以下配置切换到新 CKafka 集群消费 CKafka 集群的数据。（单个消费者消费数据，保证消息的有序性）新增消费者，需要配置 `--bootstrap-server` 中的 IP 为 CKafka 实例的接入网络：

#### 说明

如果消费者为云服务器，此处也可以继续使用原有消费者进行消费。

```
./kafka-console-consumer.sh --bootstrap-server xxx.xxx.xxx.xxx:9092 --from-beginning --new-consumer --topic topicName --consi
```

4. 切换后的消费者持续消费 CKafka 集群中的数据，迁移完毕（如果消费者为云服务器，此处也可以继续使用原有消费者进行消费）。

**注意**

上文给出的是测试命令，正式业务的运行只需要修改相应应用程序配置的 broker 地址，然后重启相应的应用即可。

## 方案3：使用 Mirrormaker 迁移

Last updated: 2021-11-15 11:37:38

## 操作场景

该任务为您介绍如何利用 MirrorMaker 将自建 Kafka 集群的数据迁移到 CKafka 中。

Kafka 的 MirrorMaker 工具可以实现将自建 Kafka 集群中的数据备份到 CKafka 集群中，具体原理如下：

MirrorMaker 可以使用一个 Consumer 从自建 Kafka 集群中消费消息，然后通过一个 Producer 将这些数据发送到 CKafka 集群中，最后您将客户端的生产消费配置转到云上实例的接入网络，即可完成从自建 Kafka 集群到 CKafka 集群的数据迁移。

## 前提条件

- 已 [购买云上 CKafka 实例](#)。
- 已 [迁移 Topic 上云](#)。

## 操作步骤

1. [下载 MirrorMaker 工具](#) 并解压到本地。

### 说明

本文以 `kafka_2.11-1.1.1.tgz` 为例。

2. 配置 `consumer.properties` 文件。

```
# list of brokers used for bootstrapping knowledge about the rest of the cluster
# format: host1:port1,host2:port2 ...
bootstrap.servers=localhost:9092

# consumer group id
group.id=test-consumer-group

partition.assignment.strategy=org.apache.kafka.clients.consumer.RoundRobinAssignor
# What to do when there is no initial offset in Kafka or if the current
# offset does not exist any more on the server: latest, earliest, none
#auto.offset.reset=
```

参数	说明
<code>bootstrap.servers</code>	自建实例的 broker 接入点列表。
<code>group.id</code>	迁移数据时用到的消费者组 ID，请勿与自建实例已存在的消费者命名重复冲突。
<code>partition.assignment.strategy</code>	分区分配的策略，以 <code>partition.assignment.strategy=org.apache.kafka.clients.consumer.RoundRobinAssignor</code> 为例。

3. 配置 `producer.properties` 文件。

```
# list of brokers used for bootstrapping knowledge about the rest of the cluster
# format: host1:port1,host2:port2 ...
bootstrap.servers=localhost:9092

# specify the compression codec for all data generated: none, gzip, snappy, lz4
compression.type=none
```

参数	说明
<code>bootstrap.servers</code>	云上实例的接入网络，在控制台的实例详情页面 <a href="#">接入方式</a> 模块的网络列复制。

接入方式 <sup>?</sup>		添加路由策略	
接入类型	接入方式	网络	操作
VPC网络	PLAINTEXT	172.17.0.1:9092	删除

compression.type	数据压缩类型，CKafka 不支持 GZip 压缩格式。
------------------	------------------------------

4. 在 `.bin` 目录下启动 `mirrormaker` 迁移工具开始迁移。

```
sh bin/kafka-mirror-maker.sh --consumer.config config/consumer.properties --producer.config config/producer.properties --whitelis
```

#### 说明

`whitelist` 为正则表达式，迁移匹配正则名称的 Topic。

5. 在 `.bin` 目录下运行 `kafka-consumer-groups.sh` 查看自建集群消费进度。

```
bin/kafka-consumer-groups.sh --new-consumer --describe --bootstrap-server 自建集群接入点 --group test-consumer-group
```

#### 说明

`group` 指迁移数据时用到的消费者组 ID。

```
[root@W-0-36-centos ~]# /usr/local/services/kafka_2.11-1.1.1/bin/kafka-consumer-groups.sh --bootstrap-server 10.0.0.36:9092 --describe --group mm1-migrate-ckafka
Note: This will not show information about old Zookeeper-based consumers.
TOPIC          PARTITION  CURRENT-OFFSET  LOG-END-OFFSET  LAG           CONSUMER-ID                                     HOST
mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
test26        0          -              461             -             mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
test193      0          -              460             -             mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
test89       0          -              461             -             mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
test114     0          -              461             -             mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
test1       3          -              2251            -             mm1-migrate-ckafka-0-598a37fa-d5f1-4ad4-ba7c-579b8c7a6d5e /10.0
```

## 后续处理

数据迁移完成后，将客户端的生产消费配置转到云上实例的接入点。

## 方案4：迁移未消费数据

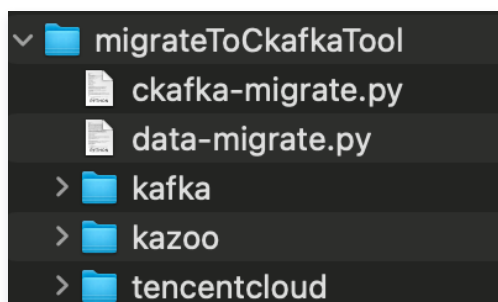
Last updated: 2023-04-21 09:44:57

### 简介

客户在迁移上云过程中，需要把旧集群上未消费完的消息迁移到新集群对应的 topic 中时。可参考本教程进行操作，即可将旧集群的未消费数据同步到新集群中。

### 前提条件

1. 保证原集群所有消费/生产已停止。
2. 保证原集群待迁移的消息保留时间足够长，即在迁移过程中避免topic消息过期自动删除。
3. 同时迁移脚本是 python 脚本，需要安装 python2，且 python2 版本>2.7.1，推荐2.7.5
4. 下载迁移工具 [migrateToCkafkaTool](#)。工具包目录如下，进入 migrateToCkafkaTool 目录下，修改 data-migrate.py 文件的配置后，执行 python data-migrate.py 即可



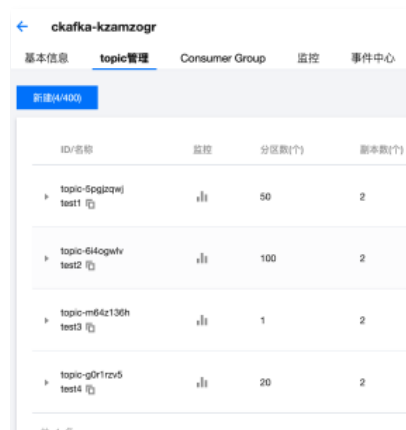
### 工具原理

脚本会扫描老集群的所有 group 列表，并取出 group 订阅中的且仍有未消费消息的 topic 列表。脚本将取出未消费完 group 订阅 topic 的 **group 提交位置** 和 **topic 末端位置**(如果一个 topic 被多个 group 订阅，那么 group 提交位置将取最小的那个)。然后将此区间位置的消息消费后再生产到新集群的对应 topic 分区中。

### 操作演示

#### 1. 在目标集群新建对应的 Topic

假设原集群是：ckafka-47bd7goz，目标新集群是：ckafka-kzamzogr。如下图所示：新集群已经建好了相同分区数的 topic。即 test1,test2,test3,test4。



原集群 ckafka-47bd7goz 有两个 group，test123-group 和 test34-group，它们分别订阅主题 test1,test2,test3 和 test3,test4。

消费组名称	状态	协议类型	均衡算法	操作
test123-group	Empty	consumer	-	offset设置 查看消费者详情 删除

分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作
test2 partition-26	1807	2669	762	查看详情
test3 partition-21	1800	2533	733	查看详情
test1 partition-34	1884	2680	796	查看详情
partition-20	1871	2629	758	查看详情
partition-60	1873	2653	780	查看详情
partition-5	1876	2679	803	查看详情
partition-15	1854	2609	755	查看详情

## 2. 下载工具包

下载迁移工具后，打开脚本填入原集群和新集群地址配置后，checkFlag 设置为0，运行脚本先预检查一下将要迁移的 topic 和位置。

```
# your local broker ip:port list
# 自建实例的broker列表 ["broker1:port1","ip2:port2"]
bootstrapServers = ["..."] # 原实例

# your cloud ckafka instance address
# 云上实例的地址 "x.x.x.x:9092"
instanceAddress = "..." # 新实例

# A dedicated group for migration data, default migrateLocalToCkafkaGroup, can be customized but should not have
# the same name as the existing consumer group in the cluster, do not delete it after the self-built cluster is
# generated, otherwise the migration progress will be lost
# 迁移数据专用group,默认migrateLocalToCkafkaGroup,可自定义但勿与集群原先已存在的消费组同名,在自建集群生成后勿删除,否则丢失迁移进度
migrateGroupId = "migrateLocalToCkafkaGroup"

# topic regex, just migrate match topics
# topic名称正则表达式,默认空迁移全部topic,非空则只迁移匹配到的topic
topicRegex = ""

# The mapping between the local topic and the cloud topic, if not set, the default local and cloud have the same name
# 本地Topic和云上Topic的映射,如果未设置则默认本地和云上同名
# example: localTopicAndCloudTopicMap = {"topic1-local": "topic1-cloud", "topic2-local": "topic2-cloud"}
localTopicAndCloudTopicMap = {}

# if you make sure the migrate topic List, please modify checkFlag = 1
# 检查标记,设0只显示将要迁移的topic列表不做真正迁移,请先以0运行检查将要迁移的topic列表,确认无误后修改为1开始迁移
# 0:列出迁移topic分区位点数据列表后脚本终止
# 1:列出迁移topic分区位点数据列表并开始迁移
checkFlag = 0 # 先设置为0运行一遍脚本检查,检查无误后修改为1再次运行脚本

logFileName = "data-migrate"
```

运行脚本后，将输出一些信息，同时当前目录会同时写入一份文本日志。

```
ckafka-migrate.py data-migrate_220914_12:09:56.log data-migrate.py kafka kazoo tencentcloud
```

## 3. 查看输出信息

通过屏幕输出或者文本日志文件检查 Prepare to migrate 的信息，这是将要迁移的位点信息。

```
-----Topic: [test3] Partition: (0) migrate-----
Prepare to migrate local-topic: [test3] partition: 0 startOffset: 187800 ~ endOffset: 264322 data to cloud-topic: [test3]

-----Topic: [test2] Partition: (0) migrate-----
Prepare to migrate local-topic: [test2] partition: 0 startOffset: 1873 ~ endOffset: 2665 data to cloud-topic: [test2]

-----Topic: [test2] Partition: (1) migrate-----
Prepare to migrate local-topic: [test2] partition: 1 startOffset: 1860 ~ endOffset: 2621 data to cloud-topic: [test2]

-----Topic: [test2] Partition: (2) migrate-----
Prepare to migrate local-topic: [test2] partition: 2 startOffset: 1830 ~ endOffset: 2591 data to cloud-topic: [test2]
```

以test3为例，它同时被 test123-group 和 test34-group 订阅，检查原集群的订阅情况。

分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作
test2 partition-0	187800	264322	76522	查看详情

分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作
test4	239119	264322	25203	查看详情
test3				查看详情

按照预定逻辑，一个 topic 被多个 group 订阅应该从提交最小的那个位置开始同步，即187800，检查输出信息与预期一致。

```
-----Topic:[test3] Partition:(0) migrate-----
Prepare to migrate local-topic:[test3] partition:0 startOffset:187800 ~ endOffset:264322 data to cloud-topic:[test3]
```

还有一种情况是原集群主题 test1 由于消息已经过期，但是 group 的提交位置在过期消息的区间，因此同步只会从 test1 还未过期的最早消息位置同步。

分区名称	leader	副本	ISR	起始offset	承诺offset	消息数	未同步副本
partition-0	100250	100250,100248	100250,100248	5226	5226	0	

分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作
test2	3706	5261	1556	查看详情
test3				查看详情
test1	3713	5226	1513	查看详情

以 test1 的0分区为例，脚本会提示 test1 主题0分区的5226位置（topic 存活消息最小位置）已经超过 group 订阅的提交 offset 的3713位置（该位置的消息已过期），因此同步开始的位置设置到了5226。又由于5226同样也是该分区目前最大的 offset（该分区目前存活消息总数为0）代表无消息可迁移，因此输出 skip migrate...的文本信息，代表跳过迁移本分区的数据。

```
-----Topic:[test1] Partition:(0) migrate-----
the local-topic:[test1] partition:0 topic_beginOffset:5226 large than group_startOffset:3713,and reset startOffset to 5226
skip migrate local-topic:[test1] partition:0 startOffset:5226 ~ endOffset:5226 data to cloud-topic:[test1],because startOffset is equal or more than endOffset

-----Topic:[test1] Partition:(1) migrate-----
the local-topic:[test1] partition:1 topic_beginOffset:5340 large than group_startOffset:3784,and reset startOffset to 5340
skip migrate local-topic:[test1] partition:1 startOffset:5340 ~ endOffset:5340 data to cloud-topic:[test1],because startOffset is equal or more than endOffset
```

#### 4. 开始迁移

经上步检查过输出的信息确认无误后，修改 checkFlag=1 开始迁移。

```
-----Topic:[test3] Partition:(0) migrate-----
Prepare to migrate local-topic:[test3] partition:0 startOffset:187800 ~ endOffset:264322 data to cloud-topic:[test3]
begin migrate local-topic:[test3] partition:0 startOffset:187800 ~ endOffset:264322 data to cloud-topic:[test3]
finish migrate local-topic:[test3] partition:0 startOffset:187800 ~ endOffset:264322 data to cloud-topic:[test3]

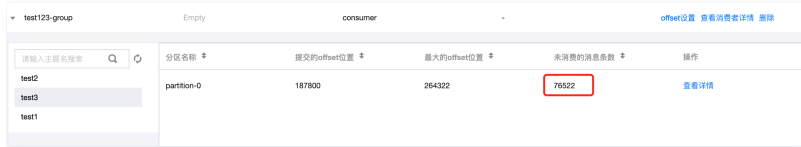
-----Topic:[test2] Partition:(0) migrate-----
Prepare to migrate local-topic:[test2] partition:0 startOffset:1873 ~ endOffset:2665 data to cloud-topic:[test2]
begin migrate local-topic:[test2] partition:0 startOffset:1873 ~ endOffset:2665 data to cloud-topic:[test2]
finish migrate local-topic:[test2] partition:0 startOffset:1873 ~ endOffset:2665 data to cloud-topic:[test2]

-----Topic:[test2] Partition:(1) migrate-----
Prepare to migrate local-topic:[test2] partition:1 startOffset:1860 ~ endOffset:2621 data to cloud-topic:[test2]
begin migrate local-topic:[test2] partition:1 startOffset:1860 ~ endOffset:2621 data to cloud-topic:[test2]
finish migrate local-topic:[test2] partition:1 startOffset:1860 ~ endOffset:2621 data to cloud-topic:[test2]

-----Topic:[test2] Partition:(2) migrate-----
Prepare to migrate local-topic:[test2] partition:2 startOffset:1830 ~ endOffset:2591 data to cloud-topic:[test2]
begin migrate local-topic:[test2] partition:2 startOffset:1830 ~ endOffset:2591 data to cloud-topic:[test2]
finish migrate local-topic:[test2] partition:2 startOffset:1830 ~ endOffset:2591 data to cloud-topic:[test2]
```

#### 5. 检查迁移后数据是否数量一致

以 test3 为例,预期迁移 test123-group 未消费的76522条消息,已经全部成功写入新集群的test3主题中,迁移数据完成。



分区名称	提交的offset位置	最大的offset位置	未消费的消息条数	操作
partition-0	187800	264392	76522	<a href="#">查看详情</a>

# 集群备份

Last updated: 2023-04-14 09:29:49

## 操作场景

针对业务连续服务、数据可靠性有强需求或是监管需要的场景，CKafka 专业版提供集群备份功能，帮助用户提升业务连续服务的能力，同时提升数据的可靠性。集群备份功能支持实例级别同步数据，支持不同地域实例间的复制迁移功能，实例的所有数据和元数据都会同步。

### 说明

当前仅 CKafka 专业版实例支持集群备份能力，标准版不支持。

## 前提条件

在创建数据同步任务之前，您需要提前创建好目标实例与 Topic。

## 操作步骤

1. 登录 [CKafka 控制台](#)。
2. 在左侧导航栏选择实例列表，单击目标实例的“ID”，进入实例基本信息页面。
3. 在页面上方选择集群备份页签，单击新建任务。
4. 在新建任务弹窗中填写任务名称。
5. 单击下一步，确认数据源信息，设置好 topic offset。

### 说明：

目标实例 Topic 与源 Topic 分区需保持一致。如调整分区，同步程序会在分区一致后，恢复同步。

新建任务

任务基本信息 > 2 选择数据源 > 3 选择数据目标

源实例地域 广州

源实例 ckafka-pk

仅限专业版实例，目标实例topic与源topic分区需保持一致。  
如调整分区，同步程序会在分区一致后，恢复同步。

起始位置①  从最新位置开始消费  
 从最开始位置开始消费  
 从时间点位置开始消费

上一步 下一步

6. 单击下一步，选择目标实例地域、目标实例，设置好 topic offset。

新建任务

任务基本信息 > 选择数据源 > 3 选择数据目标

目标实例地域 广州

目标实例 ckafka-

仅限专业版实例，目标实例topic与源topic分区需保持一致。  
如调整分区，同步程序会在分区一致后，恢复同步。

上一步 提交

7. 单击提交，在数据同步页面可看到创建的跨地域容灾任务。

### 说明：

任务创建成功后，会自动开始数据同步，并且数据是实时复制的。您可参见 [查看任务进度](#) 查看数据同步进度。

## 查看任务进度

前提条件：已创建数据同步任务。

1. 在集群备份页面，单击目标任务的“ID”，进入任务基本信息页面。

2. 选择同步进度页签，可查看数据同步的进度。

- **数据同步**：展示各 Topic 的同步进度。

ID/名称	同步状态
topic-...	同步中
ckaika-topic-demo	同步中
topic-...	同步中
create	同步中
topic-...	同步中
rs	同步中
topic-...	同步中
say	同步中
inter-topic-test	同步中

- **元数据同步**：展示 Topic、ACL 策略、用户和 Consumer Group 的同步进度。

资源名称	同步进度
topic	已同步 11/11 个
ACL策略	已同步 1/1 个
用户	已同步 4/4 个
Consumer Group	已同步 0/0 个

共 4 条

## 暂停任务

### 说明

处于运行状态的任务可以暂停，当您发现数据同步服务影响了 CKafka 正常服务时，可以暂停数据同步。

在集群备份页面，单击目标任务的操作栏的暂停，可暂停任务。

## 恢复任务

### 说明

处于暂停同步状态的任务可以重新启动，将继续同步数据，且数据将从暂停位置继续同步。

在集群备份页面，单击目标任务的操作栏的恢复，可将暂停任务恢复。

## 删除任务

### 注意

- 删除任务表示停止数据同步，不会影响到已经同步的数据和相关的 CKafka 实例。
- 任务一旦删除不可恢复，请您谨慎操作。

在集群备份页面，单击目标任务的操作栏的删除，可删除任务。

# 数据压缩

Last updated: 2023-06-28 14:15:01

## 操作场景

数据压缩可以减少网络 IO 传输量，减少磁盘存储空间。您可以通过本文档，了解数据压缩支持的消息格式，并根据需求配置数据压缩。

## 消息格式

目前 CKafka 支持两类消息格式，分别为 V1 版本和 V2 版本（在 0.11.0.0 引入）。目前 CKafka 支持 0.9、0.10、1.1、2.4 和 2.8 版本。不同版本对应不同的配置，说明如下：

- 消息格式转换主要是为了兼容老版本的消费者程序，在一个 CKafka 集群中通常同时保存多种版本的消息格式（V1/V2）。
- Broker 端会对新版本消息执行向老本格式的转换，该过程中会涉及消息的解压缩和重新压缩。
- 消息格式转换对性能的影响很大，除了增加额外的压缩和解压缩操作之外，还会让 CKafka 丧失其优秀的零拷贝（Zero-copy）特性。因此，一定要保证消息格式的统一。
- 零拷贝（Zero-copy）：数据在磁盘和网络进行传输时，避免昂贵的内核态数据拷贝，从而实现快速的数据传输。

## 压缩算法对比

官方推荐使用的压缩算法为 Snappy 算法。分析过程如下：

评估一个压缩算法的优劣，主要有两个指标：压缩比、压缩/解压缩吞吐量。

CKafka 2.1.0 之前的版本支持三种压缩算法：GZIP、Snappy、LZ4。

在 CKafka 的实际使用中，三种算法的性能指标对比如下：

- 压缩比：LZ4 > GZIP > Snappy
- 吞吐量：LZ4 > Snappy > GZIP

物理资源占用如下：

- 带宽：由于 Snappy 的压缩比最低，因此占用的网络带宽最大。
- CPU：各个压缩算法差不多，在压缩时 Snappy 使用更多的 CPU，在解压缩时 GZIP 使用更多的 CPU。

因此，正常情况下三种压缩算法的推荐排序为：LZ4 > GZIP > Snappy。

经过长时间的现网运行试验，发现在大多数情况下上面的模型是没问题的。但是在某些极端情况下 LZ4 压缩算法会导致 CPU 负载增大。

经分析是业务的源数据内容不一样，导致压缩算法的性能表现不一样。故建议对 CPU 指标敏感的用户采用更为稳定的 Snappy 压缩算法。

## 配置数据压缩

生产者可通过下述方法配置数据压缩：

```
Properties props = new Properties();
props.put("bootstrap.servers", "localhost:9092");
props.put("acks", "all");
props.put("key.serializer", "org.apache.kafka.common.serialization.StringSerializer");
props.put("value.serializer", "org.apache.kafka.common.serialization.StringSerializer");

// Producer 启动后，生产的每个消息集合都会经过压缩，能够很好地节省网络传输带宽和 Kafka Broker 端的磁盘占用

// 请注意不同版本对应不同的配置，0.9及以下版本不允许使用压缩。0.10及以上版本不允许使用 GZip 压缩。

props.put("compression.type", "lz4");
Producer<String, String> producer = new KafkaProducer<>(props);
```

大部分情况下，Broker 从 Producer 接收到消息后，仅仅只是原封不动地保存，而不会对其进行任何修改

## 说明与注意

- 发送数据到 CKafka，不能设置压缩 compression.codec。
- 默认不支持 Gzip 压缩格式，如果需要支持，请 [提交工单](#) 申请。  
Gzip 压缩对于 CPU 的消耗较高，使用 Gzip 会导致所有的消息都是 Invalid 消息。
- 使用 LZ4 压缩方法时，程序不能正常运行，可能的原因如下：  
消息格式错误。CKafka 默认版本为 0.10.2，您需要使用 V1 版本的消息格式。

- 不同 CKafka Client 的 SDK 设置方式不同，您可以通过开源社区进行查询（例如 [C/C++ Client 的说明](#)），设置消息格式的版本。