

分布式消息服务 Kafka

用户指南

文档版本 01
发布日期 2025-09-17



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

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

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

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

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

目 录

- 1 Kafka 业务使用流程..... 1
- 2 权限管理.....3
 - 2.1 创建用户并授权使用 DMS for Kafka..... 3
- 3 购买 Kafka 实例..... 7
- 4 配置 Topic.....25
 - 4.1 创建 Kafka Topic..... 25
 - 4.2 设置 Kafka Topic 权限..... 31
 - 4.3 Topic 管理.....37
 - 4.3.1 查看 Kafka Topic 详情..... 37
 - 4.3.2 查看 Kafka Topic 日志.....40
 - 4.3.3 修改 Kafka Topic 配置.....43
 - 4.3.4 修改 Kafka Topic 分区数..... 46
 - 4.3.5 修改 Kafka Topic 副本数..... 50
 - 4.3.6 导出 Kafka Topic 列表..... 54
 - 4.3.7 修改 Kafka 分区平衡.....54
 - 4.3.8 配置 Kafka 自动创建 Topic..... 66
 - 4.3.9 删除 Kafka Topic.....67
- 5 连接实例.....70
 - 5.1 配置 Kafka 网络连接..... 70
 - 5.1.1 连接 Kafka 网络要求..... 70
 - 5.1.2 配置 Kafka 实例的公网访问..... 72
 - 5.1.3 使用 VPCEP 实现跨 VPC 访问 Kafka..... 78
 - 5.1.4 使用 DNAT 实现公网访问 Kafka..... 84
 - 5.2 配置 Kafka 访问控制..... 89
 - 5.2.1 配置 Kafka 实例的明文/密文接入..... 89
 - 5.2.2 制作和替换 Kafka JKS 格式的 SSL 证书..... 93
 - 5.2.3 获取和使用 Kafka PEM 格式的 SSL 证书..... 102
 - 5.2.4 配置 Kafka 的 SSL 双向认证..... 103
 - 5.2.5 配置 Kafka ACL 用户..... 111
 - 5.3 配置 Kafka 客户端..... 114
 - 5.3.1 Kafka 客户端参数配置建议..... 114
 - 5.3.2 Kafka 客户端使用建议..... 119

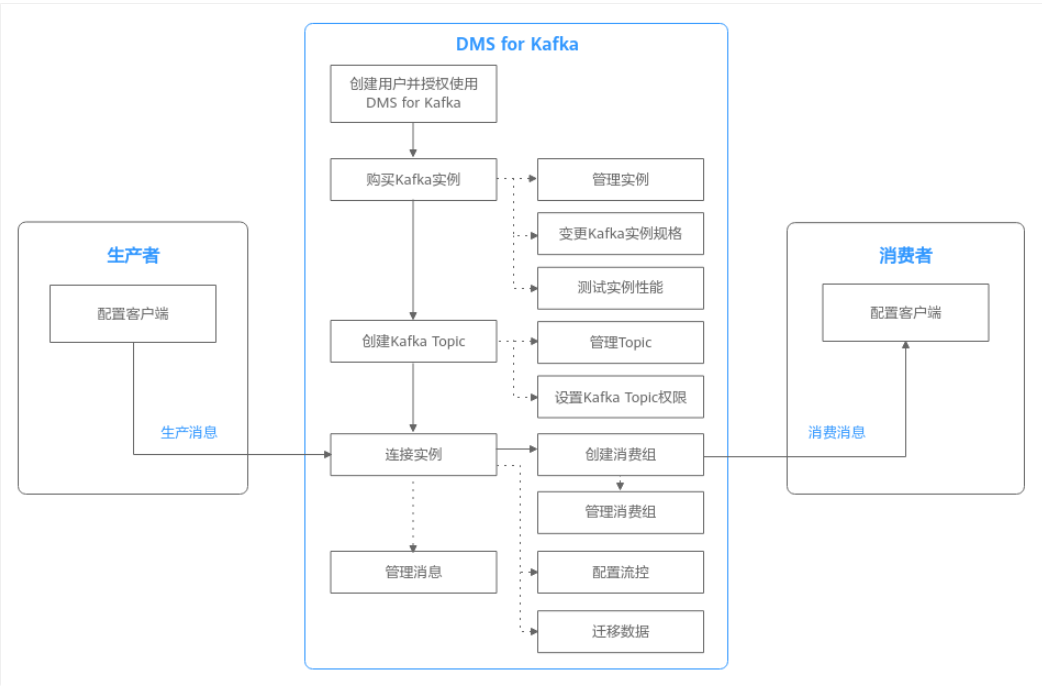
5.4 使用客户端连接 Kafka（明文接入）	120
5.5 使用客户端连接 Kafka（密文接入）	123
5.6 使用控制台连接 Kafka.....	130
6 管理消息.....	132
6.1 查看 Kafka 消息.....	132
6.2 修改 Kafka 消息老化时间.....	135
6.3 删除 Kafka 消息.....	137
6.4 诊断 Kafka 消息积压.....	139
7 管理消费组.....	142
7.1 创建 Kafka 消费组.....	142
7.2 查看 Kafka 消费组信息.....	144
7.3 查看 Kafka 消费者信息.....	145
7.4 查看和重置 Kafka 消费进度.....	149
7.5 查看 Kafka 重平衡日志.....	153
7.6 修改 Kafka 消费组信息.....	156
7.7 配置 Kafka 自动创建消费组.....	156
7.8 导出 Kafka 消费组列表.....	157
7.9 删除 Kafka 消费组.....	157
7.10 取消 Kafka 消费组和 Topic 的订阅关系.....	159
8 配置流控.....	161
8.1 配置 Kafka 流控.....	161
8.2 查看 Kafka 流控统计.....	166
9 管理实例.....	168
9.1 查看和修改 Kafka 实例基本信息.....	168
9.2 查看 Kafka 磁盘使用量.....	173
9.3 查看 Kafka 实例后台任务.....	174
9.4 查看 Kafka 生产消费示例代码.....	175
9.5 修改 Kafka 实例配置参数.....	175
9.6 配置 Kafka 实例标签.....	180
9.7 配置 Kafka 实例回收站策略.....	182
9.8 升级 Kafka 实例内核版本.....	184
9.9 导出 Kafka 实例列表.....	186
9.10 重启 Kafka 实例.....	187
9.11 删除 Kafka 实例.....	187
9.12 管理 Kafka Manager.....	188
9.12.1 访问 Kafka Manager.....	188
9.12.2 重置 Kafka Manager 密码.....	194
9.12.3 重启 Kafka Manager.....	195
9.12.4 关闭 Kafka Manager.....	196
10 变更实例规格.....	197

10.1 变更 Kafka 集群实例规格.....	197
10.2 自动扩容 Kafka 实例存储空间.....	204
11 迁移数据.....	206
11.1 Kafka 数据迁移概述.....	206
11.2 使用 MirrorMaker 跨集群同步数据.....	210
11.3 使用 Smart Connect 迁移数据.....	213
11.3.1 开启 Smart Connect.....	213
11.3.2 配置 Kafka 间的数据复制.....	214
11.3.3 配置 Kafka 数据转储至 OBS.....	221
11.3.4 管理 Smart Connect 任务.....	223
11.3.5 关闭 Smart Connect.....	225
12 测试实例性能.....	227
12.1 测试 Kafka 生产速率和 CPU 消耗.....	227
12.2 测试 Kafka 实例 TPS.....	240
13 申请扩大 Kafka 配额.....	247
14 查看监控指标与配置告警.....	249
14.1 查看 Kafka 监控数据.....	249
14.2 Kafka 支持的监控指标.....	252
14.3 配置 Kafka 监控告警.....	264
15 查看 Kafka 审计日志.....	268

1 Kafka 业务使用流程

分布式消息服务Kafka版是一款基于开源社区版Kafka提供的消息队列服务，向用户提供计算、存储和带宽资源独占式的Kafka实例。使用Kafka实例生产消费消息的流程如下图所示。

图 1-1 Kafka 业务使用流程



1. **创建用户并授权使用DMS for Kafka**
创建IAM用户，并授予DMS for Kafka的权限，以达到用户之间权限的隔离。
2. **购买Kafka实例**
Kafka实例采用物理隔离的方式部署，租户独占Kafka实例。
3. **创建Kafka Topic**
创建用于存储消息的Topic，供生产者生产消息以及消费者订阅消息。
4. **连接实例**
客户端使用内网/公网连接Kafka实例，并生产消费消息。

2 权限管理

2.1 创建用户并授权使用 DMS for Kafka

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

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

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

本章节为您介绍对用户授权的方法，操作流程如[图2-1](#)所示。

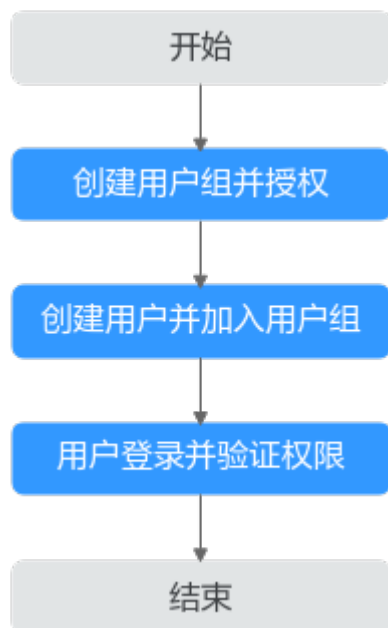
前提条件

给用户组授权之前，请您了解用户组可以添加的DMS for Kafka系统策略，并结合实际需求进行选择，DMS for Kafka支持的系统策略及策略间的对比，请参见：[DMS for Kafka系统策略](#)。若您需要对除DMS for Kafka之外的其它服务授权，IAM支持服务的所有策略请参见[权限策略](#)。

DMS for Kafka的权限与策略基于分布式消息服务DMS，因此在IAM服务中为用户组授予DMS for Kafka权限时，请选择并使用“DMS”的权限与策略。

示例流程

图 2-1 给用户授权 DMS for Kafka 权限流程



1. 创建用户组并授权

在IAM控制台创建用户组，并授予DMS for Kafka的只读权限“DMS ReadOnlyAccess”。

2. 创建用户并加入用户组

在IAM控制台创建用户，并将其加入1中创建的用户组。

3. 用户登录并验证权限

新创建的用户登录到控制台，切换至授权区域，验证权限：

- 在“服务列表”中选择分布式消息服务Kafka版，进入Kafka实例主界面，单击右上角“购买Kafka实例”，尝试购买Kafka实例，如果无法购买Kafka实例（假设当前权限仅包含DMS ReadOnlyAccess），表示“DMS ReadOnlyAccess”已生效。
- 在“服务列表”中选择云硬盘（假设当前策略仅包含DMS ReadOnlyAccess），若提示权限不足，表示“DMS ReadOnlyAccess”已生效。
- 在“服务列表”中选择分布式消息服务Kafka版（假设当前策略仅包含DMS ReadOnlyAccess），进入Kafka实例主界面，如果能够查看Kafka实例列表，表示“DMS ReadOnlyAccess”已生效。

DMS for Kafka 自定义策略样例

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

目前华为云支持以下两种方式创建自定义策略：

- 可视化视图创建自定义策略：无需了解策略语法，按可视化视图导航栏选择云服务、操作、资源、条件等策略内容，可自动生成策略。

- JSON视图创建自定义策略：可以在选择策略模板后，根据具体需求编辑策略内容；也可以直接在编辑框内编写JSON格式的策略内容。

具体创建步骤请参见：[创建自定义策略](#)。下面为您介绍常用的DMS for Kafka自定义策略样例。

- 示例1：授权用户删除实例和重启实例

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

- 示例2：拒绝用户删除实例

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

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

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

DMS for Kafka 资源

资源是服务中存在的对象。在DMS for Kafka中，资源包括：kafka，您可以在创建自定义策略时，通过指定资源路径来选择特定资源。

表 2-1 DMS for Kafka 的指定资源与对应路径

指定资源	资源名称	资源路径
kafka	实例	【 格式 】 DMS:*:*kafka:实例ID 【 说明 】 对于实例资源，IAM自动生成资源路径前缀DMS:*:*kafka:通过实例ID指定具体的资源路径，支持通配符*。例如： DMS:*:*kafka:*表示任意Kafka实例。

DMS for Kafka 请求条件

您可以在创建自定义策略时，通过添加“请求条件”（Condition元素）来控制策略何时生效。请求条件包括条件键和运算符，条件键表示策略语句的Condition元素，分为全局级条件键和服务级条件键。[全局级条件键](#)（前缀为g:）适用于所有操作，服务级条件键（前缀为服务缩写，如dms:）仅适用于对应服务的操作。运算符与条件键一起使用，构成完整的条件判断语句。

DMS for Kafka通过IAM预置了一组条件键，例如，您可以先使用dms:ssl条件键检查Kafka实例是否开启SASL，然后再允许执行操作。下表显示了适用于DMS for Kafka服务特定的条件键。

表 2-2 DMS for Kafka 请求条件

DMS for Kafka条件键	运算符	描述
dms:connector	Bool Null	是否开启Smart Connect
dms:publicIP	Bool Null	是否开启公网
dms:ssl	Bool Null	是否开启SSL

3 购买 Kafka 实例

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

准备实例依赖资源

创建Kafka实例前，请提前准备好如表3-1所示资源。

表 3-1 Kafka 实例依赖资源

资源名称	要求	创建指导
VPC和子网	Kafka实例可以使用当前账号下已创建的VPC和子网，也可以使用新创建的VPC和子网，还可以使用共享VPC和子网，请根据实际需要进行配置。 共享VPC是基于资源访问管理（Resource Access Manager，简称RAM）服务的机制，VPC的所有者可以将VPC内的子网共享给一个或者多个账号使用。通过共享VPC功能，可以简化网络配置，帮助您统一配置和运维多个账号下的资源，有助于提升资源的管控效率，降低运维成本。有关VPC子网共享的更多信息，请参见共享VPC。 在创建VPC和子网时应注意： - 创建的VPC与Kafka实例在相同的区域。 - 子网开启IPv6后，Kafka实例支持IPv6功能。Kafka实例开启IPv6后，客户端可以使用IPv6地址连接实例。	创建VPC和子网的操作指导请参考创建虚拟私有云和子网，若需要在已有VPC上创建和使用新的子网，请参考为虚拟私有云创建新的子网。

资源名称	要求	创建指导
安全组	不同的Kafka实例可以重复使用相同的安全组，也可以使用不同的安全组，请根据实际需要进行配置。 安全组与Kafka实例必须在相同的区域。 连接Kafka实例前，请根据连接方式配置对应的安全组，具体请参考 表5-2 。	创建安全组的操作指导请参考 创建安全组 ，为安全组添加规则的操作指导请参考 添加安全组规则 。
弹性IP地址	如果客户端使用公网连接Kafka实例，请提前创建弹性IP地址。 在创建弹性IP地址时，应注意如下要求： • 创建的弹性IP地址与Kafka实例在相同的区域。 • 弹性IP地址的数量必须与Kafka实例的代理个数相同。 • Kafka实例无法识别开启IPv6转换功能的弹性IP地址。	创建弹性IP地址的操作指导请参考 申请弹性公网IP 。

约束与限制

- 开启IPv6的实例不支持动态修改SASL_SSL功能。
- 单机实例不支持密文接入和Smart Connect。

购买 Kafka 实例

分布式消息服务Kafka版控制台提供了多种购买Kafka实例的方式，满足您在不同业务场景的需求。

表 3-2 Kafka 实例购买方式

购买方式	适用场景
快速购买Kafka集群实例	快速购买Kafka实例的操作指导。分布式消息服务Kafka版已经预设了几种实例规格，根据业务需要选择对应规格进行购买。
自定义购买Kafka单机/集群实例	购买Kafka单机/集群实例的通用操作指导。根据业务场景灵活设置Kafka配置，支持购买Kafka单机/集群实例。

快速购买 Kafka 集群实例

- 步骤1 进入[购买Kafka实例页面](#)。
- 步骤2 在“快速购买”页签中，设置实例基础配置。

表 3-3 实例基础配置参数说明

参数名称	说明
计费模式	- 包年/包月：一种预付费模式，即先付费再使用，按照订单的购买周期进行结算。 - 按需付费：一种后付费模式，即先使用再付费，按照Kafka实例实际使用时长计费，秒级计费，按小时结算。
区域	不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。
可用区	可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。 根据实际情况选择1个或者3个及以上可用区。实例创建成功后，不支持修改可用区。

步骤3 设置套餐规格。

- 综合推荐：分布式消息服务Kafka版已经预设了几种规格，请根据需要选择。磁盘类型和磁盘大小根据实际需求进行设置。**Kafka实例创建后，磁盘类型不支持修改。**
存储空间用于存储消息（包括副本中的消息）、日志和元数据，建议根据业务消息体积、副本数量以及预留磁盘大小选择存储空间大小。每个Kafka代理会预留33GB的磁盘空间，用于存储日志和元数据。
创建实例时会进行磁盘格式化，磁盘格式化会导致实际可用磁盘为总磁盘的93%~95%。
磁盘类型支持高I/O、超高I/O、极速型SSD和通用型SSD，更多磁盘信息请参考[如何选择磁盘类型](#)。

图 3-1 综合推荐



- 规格测算：系统根据您选择的版本号和输入的参数（生产流量峰值、消费流量、单个Topic副本数、总分区数和老化时间内的生产消息量），为您计算出不同规格所需的代理个数和代理存储空间，您根据实际情况选择。

图 3-2 规格测算

套餐规格 ①

综合推荐

规格测算

实例创建后，代理数量 不支持缩容。若规格不符合您的需求，可跳转至 自定义购买。

版本

3.x

2.7

2.3.0

1.1.0

实例安装的软件版本。版本支持公告 版本发布说明 实例创建后，不支持变更版本。建议服务端版本和客户端版本保持一致。

生产流量峰值(MB/s)

100

计算方式：生产TPS峰值 * 消息大小

消费流量(MB/s)

100

计算方式：生产流量峰值 * 消费组数

单个Topic副本数

3

副本数默认为3，如果配置为1则业务不具有高可靠。

总分区数

300

计算方式：总Topic数 * 每个Topic的分区数

老化时间内的生产消息量(GB)

10

计算方式：老化时间 * 平均生产流量

一键测算

① 根据您填写的参数值，计算出实际存储空间，考虑到存储空间需要预留一定余量，推荐规格中的存储空间*80%为实际存储空间。

测算的推荐规格

规格名称	代理数量	单个代理存储空间	实例总存储空间
☒ kafka.2u4g.cluster.small	3	100 GB	超高I/O 300 GB
☐ kafka.2u4g.cluster.small	4	100 GB	高I/O 400 GB
☐ kafka.2u4g.cluster	3	100 GB	超高I/O 300 GB

步骤4 设置网络信息。

表 3-4 实例网络参数说明

参数名称	说明
虚拟私有云	选择已经创建好的或共享的虚拟私有云。 虚拟私有云可以为您的Kafka实例构建隔离的、能自主配置和管理的虚拟网络环境。如果您需要创建新虚拟私有云，单击右侧的“创建虚拟私有云”，弹出“创建虚拟私有云”对话框。参考创建虚拟私有云和子网，创建虚拟私有云。 虚拟私有云在Kafka实例创建完成后，不支持修改。
子网	选择已经创建好的或共享的子网。如果您需要创建新子网，单击右侧的“创建子网”，弹出“新建子网”对话框。参考为虚拟私有云创建新的子网，创建子网。 子网在Kafka实例创建完成后，不支持修改。 子网开启IPv6后，Kafka实例支持IPv6功能。
IPv6	子网开启IPv6后，页面才显示此参数。开启IPv6后，客户端可以使用IPv6地址连接实例。 开启IPv6的实例不支持动态开启和关闭SASL_SSL功能。 实例创建成功后，不支持修改IPv6开关。 此功能在“华东二”、“华南-广州”和“华东-上海一”区域上线。
内网IP地址	根据实际需求选择“自动分配”或“手动选择”。 - 自动分配：系统自动分配子网中可用的IP地址。 - 手动选择：在下拉框中勾选指定的IP地址。如果指定的IP地址数量少于代理数量时，系统会自动为剩余代理分配IP地址。 开启IPv6后，控制台不显示此参数。

参数名称	说明
安全组	选择已经创建好的安全组。 安全组是一组对Kafka实例的访问规则的集合。如果您需要创建新安全组，单击右侧的“创建安全组”，弹出“创建安全组”对话框。参考创建安全组，设置安全组参数。 客户端连接Kafka实例前，请根据不同的连接方式配置对应的安全组规则，否则会连接失败。具体的安全组规则请参考表 5-2。

步骤5 设置实例访问方式。

图 3-3 实例访问方式

访问方式

☒ 内网访问

内网接入方式

☒ 明文接入 ☐ 密文接入

实例创建成功后，跨VPC访问安全协议也将默认为 PLAINTEXT 协议，且无法修改。

☒ 公网访问

公网接入方式

☐ 明文接入 ☐ 密文接入

公网访问不开启密文接入将会有安全风险!

公网IP地址

请选择

创建弹性IP

请指定弹性IP地址以支持公网访问。应选择 3 个，已选择 0 个。

表 3-5 实例访问方式参数说明

参数名称	子参数	说明
内网访问	接入方式	接入方式分为以下两种： - 明文接入：表示客户端连接Kafka实例时，无需进行SASL认证。 - 密文接入：表示客户端连接Kafka实例时，需要进行SASL认证。 开启“密文接入”后，需要设置kafka安全协议、SSL用户名、密码和SASL PLAIN机制。 内网访问不支持关闭，明文接入和密文接入至少开启一个。 接入方式的取值决定了跨VPC访问协议的取值，具体如下： - 开启明文接入且关闭密文接入时，跨VPC访问Kafka实例需要使用“PLAINTEXT”协议。 - 开启密文接入且kafka安全协议为“SASL_SSL”时，跨VPC访问Kafka实例需要使用“SASL_SSL”协议。 - 开启密文接入且kafka安全协议为“SASL_PLAINTEXT”时，跨VPC访问Kafka实例需要使用“SASL_PLAINTEXT”协议。 实例创建成功后，跨VPC访问安全协议无法修改。
	公网访问	接入方式分为以下两种： - 明文接入：表示客户端连接Kafka实例时，无需进行SASL认证。 - 密文接入：表示客户端连接Kafka实例时，需要进行SASL认证。 开启“密文接入”后，需要设置kafka安全协议、SSL用户名、密码和SASL PLAIN机制。 开启公网访问后，明文接入和密文接入至少开启一个，否则无法连接Kafka实例。
公网访问	公网IP地址	根据页面提示信息选择对应数量的公网IP地址。 如果公网IP地址数量不足，请单击“创建弹性IP”跳转到弹性公网IP页面，创建公网IP地址。然后返回购买Kafka实例页面，在“公网IP地址”后单击，刷新公网IP地址。 Kafka实例不支持绑定IPv6的弹性公网IP，只支持绑定IPv4的弹性公网IP。

kafka安全协议、SSL用户名、密码和SASL PLAIN机制的参数解释如下：

表 3-6 密文接入参数说明

参数名称	参数值	说明
kafka安全协议	SASL_SSL	采用SASL方式进行认证，数据通过SSL证书进行加密传输，安全性更高。
	SASL_PLAINTEXT	采用SASL方式进行认证，数据通过明文传输，性能更好。 由于明文传输，建议使用SCRAM-SHA-512机制。
SSL用户名	-	客户端用于连接Kafka实例的用户名。 用户名需要符合以下命名规则：由英文字母开头，且只能由英文字母、数字、中划线、下划线组成，长度为4~64个字符。 密文接入成功开启后，用户名不支持修改。
密码	-	客户端用于连接Kafka实例的密码。 密码需要符合以下规则： • 长度为8~32个字符。 • 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符~!@#\$%^&*()-_+=\ []{};:","<.>?和空格，并且不能以-开头。 • 不能与用户名或倒序的用户名相同。
SASL认证机制	-	• 未勾选“PLAIN”时，使用SCRAM-SHA-512机制对账号密码进行认证。 • 勾选“PLAIN”后，同时支持SCRAM-SHA-512机制和PLAIN机制，根据实际情况选择其中任何一种配置连接。 密文接入成功开启后，SASL认证机制不支持修改。 什么是SCRAM-SHA-512机制和PLAIN机制？ • SCRAM-SHA-512机制：采用哈希算法对用户名与密码生成凭证，进行身份校验的安全认证机制，比PLAIN机制安全性更高。 • PLAIN机制：一种简单的用户名密码校验机制。

步骤6 设置高级配置。

表 3-7 高级配置参数说明

参数名称	说明
实例名称	名称支持自定义，但需要符合命名规则：长度为4~64个字符，由英文字母开头，只能由英文字母、数字、中划线、下划线组成。

参数名称	说明
企业项目	该参数针对企业用户使用。 企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理，默认项目为default。
容量阈值策略	设置磁盘使用达到容量阈值后的消息处理策略，容量阈值为95%。 - 自动删除：可以正常生产和消费消息，但是会删除最早的10%的消息，以保证磁盘容量充足。该场景优先保障业务不中断，数据存在丢失的风险。 - 生产受限：无法继续生产消息，但可以继续消费消息。该场景适用于对数据不能丢的业务场景，但是会导致生产业务失败。
Smart Connect	选择是否开启Smart Connect。 Smart Connect用于异构系统间的数据同步，开启Smart Connect后，设置Smart Connect任务可以实现Kafka实例和其他云服务之间的数据同步，或者两个Kafka实例之间的数据同步。 开启Smart Connect后，实例会另外创建2个用于Smart Connect的代理。
Kafka自动创建Topic	选择是否开启Kafka自动创建Topic功能。 开启Kafka自动创建Topic功能表示向一个未创建的Topic生产或消费消息时，系统会自动创建此Topic，此Topic的默认参数值如表3-8所示。 - 集群实例如果在“实例管理 > 配置参数”中修改“log.retention.hours”（老化时间）、“default.replication.factor”（副本数）或“num.partitions”（分区数）的参数值，此后自动创建的Topic参数值为修改后的参数值。例如：“num.partitions”修改为“5”，自动创建的Topic参数值如表3-8所示。 - 单机实例不支持修改配置参数。

参数名称	说明
标签	标签用于标识云资源，当您拥有相同类型的许多云资源时，可以使用标签按各种维度（例如用途、所有者或环境）对云资源进行分类。 如您的组织已经设定分布式消息服务Kafka版的相关标签策略，则需按照标签策略规则为Kafka实例添加标签。标签如果不符合标签策略的规则，则可能会导致Kafka实例创建失败，请联系组织管理员了解标签策略详情。 - 如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键值对。另外，您可以单击“创建预定义标签”，跳转到标签管理服务页面，查看已经预定义的标签，或者创建新的标签。 - 您也可以直接在“标签键”和“标签值”中设置标签。 当前每个Kafka实例最多支持设置20个不同标签，标签的命名规格，请参考配置Kafka实例标签章节。
描述	设置实例的描述信息，长度为0~1024个字符。

表 3-8 Topic 参数值

参数名称	单机实例默认参数值	集群实例默认参数值	集群实例修改后的参数值
分区数	1	3	5
副本数	1	3	3
老化时间	72小时	72小时	72小时
同步复制	不开启	不开启	不开启
同步落盘	不开启	不开启	不开启
消息时间戳类型	CreateTime	CreateTime	CreateTime
批处理消息最大值	10485760字节	10485760字节	10485760字节

步骤7 设置实例购买时长。

当选择了“包年/包月”付费模式时，页面才显示“购买时长”参数，您可以根据业务需要选择购买时长。如果勾选“自动续费”，在“包年/包月”资源到期后，自动进行续费。

- 按月购买的实例：自动续费周期为1个月。
- 按年购买的实例：自动续费周期为1年。

步骤8 单击“确认订单”，进入“确认配置”页面。

步骤9 确认实例信息无误且阅读并同意《华为云用户协议》后，如果“计费模式”选择“包年/包月”，单击“去支付”。根据界面提示信息，支付费用。如果“计费模式”选择“按需付费”，单击“提交”。

步骤10 在实例列表页面，查看Kafka实例是否创建成功。

创建实例大约需要3到15分钟，此时实例的“状态”为“创建中”。

- 当实例的“状态”变为“运行中”时，说明实例创建成功。
- 当实例的“状态”变为“创建失败”，请参考[删除Kafka实例](#)，删除创建失败的实例，然后重新创建。如果重新创建仍然失败，请联系客服。

 说明

创建失败的实例，不会占用其他资源。

----结束

自定义购买 Kafka 单机/集群实例

步骤1 进入[购买Kafka实例页面](#)。

步骤2 在“自定义购买”页签中，设置实例基础配置。

表 3-9 实例基础配置参数说明

参数名称	说明
计费模式	• 包年/包月：一种预付费模式，即先付费再使用，按照订单的购买周期进行结算。 • 按需付费：一种后付费模式，即先使用再付费，按照Kafka实例实际使用时长计费，秒级计费，按小时结算。
区域	不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。
可用区	可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。 根据实际情况选择1个或者3个及以上可用区。实例创建成功后，不支持修改可用区。

步骤3 设置实例规格信息。

表 3-10 实例规格参数说明

参数名称	说明
版本	Kafka的版本号，支持1.1.0、2.7、3.x。 实例创建后，版本号不支持修改。
部署架构	根据需求选择“单机”或“集群”。 仅2.7版本支持创建单机实例。关于单机实例和集群实例的差异，请参见 单机和集群Kafka实例差异概述 。
代理规格	请根据业务需求选择相应的代理规格。 单个代理最大分区数*代理数量=实例分区数上限。当所有Topic的总分区数大于实例分区数上限时，创建Topic失败。

参数名称	说明
代理数量	选择代理数量。
磁盘类型	选择存储Kafka数据的磁盘类型。 Kafka实例创建后，磁盘类型不支持修改。
单个代理磁盘容量	选择存储Kafka数据的磁盘大小。 存储空间用于存储消息（包括副本中的消息）、日志和元数据，建议根据业务消息体积、副本数量以及预留磁盘大小选择存储空间大小。每个Kafka代理会预留33GB的磁盘空间，用于存储日志和元数据。 创建实例时会进行磁盘格式化，磁盘格式化会导致实际可用磁盘为总磁盘的93%~95%。 磁盘类型支持高I/O、超高I/O、极速型SSD和通用型SSD，更多磁盘信息请参考如何选择磁盘类型。

图 3-4 实例规格

自定义购买

版本

3.x

2.7

2.3.0

1.1.0

实例安装的软件版本。[版本支持公告](#) [版本发布说明](#) 实例创建后，不支持变更版本。建议服务端版本和客户端版本保持一致。

部署架构

单机

集群

单机版实例创建后不支持变更为集群版。

代理规格

规格名称	单个代理TPS	单个代理最大分区数	单个代理客户端总连接数上限	单个代理消费组数	单个代理带宽（MB/s）
☒ kafka-2u4g-cluster-small	20,000	100	2,000	15	40
☐ kafka-2u4g-cluster	30,000	250	2,000	20	100
☐ kafka-4u8g-cluster	100,000	500	4,000	100	200
☐ kafka-8u16g-cluster	150,000	1,000	4,000	150	375
☐ kafka-12u24g-cluster	200,000	1,500	4,000	200	625
☐ kafka-16u32g-cluster	250,000	2,000	4,000	200	750

已选规格：规格名称： kafka-2u4g-cluster-small | 单个代理TPS： 20,000 | 单个代理最大分区数： 100 | 单个代理客户端总连接数上限： 2,000 | 单个代理消费组数： 15 | 单个代理带宽（MB/s）： 40

代理数量

-

3

+

当前实例网络总流量规格： 300 MB/s；当前实例业务流量规格： [建议读流量 50 MB/s，建议写流量 50 MB/s](#)。

流量提示：实例网络总流量规格和业务流量规格参考说明请参考 [实例规格和网络带宽说明](#)，由于一般业务系统流量存在波动，建议您预留 30% 冗余流量，防止出现流量波动影响您的业务。

磁盘类型

极速型SSD

单个代理磁盘容量

-

100

+

当前实例总存储空间为 300 GB（代理数量 3 个 * 单个磁盘容量 100 GB）。当前单个代理磁盘带宽为：170MB/s，实例磁盘总带宽规格为：510MB/s（单个代理磁盘规格 * 代理数量）。实例创建完成后，[磁盘类型不可修改](#)，[存储空间不支持扩容](#)，所选磁盘性能和开发请参考 [如何选择磁盘类型](#)，并根据实际业务IO要求进行选择。

步骤4 设置网络信息。

表 3-11 实例网络参数说明

参数名称	说明
虚拟私有云	选择已经创建好的或共享的虚拟私有云。 虚拟私有云可以为您的Kafka实例构建隔离的、能自主配置和管理的虚拟网络环境。如果您需要创建新虚拟私有云，单击右侧的“创建虚拟私有云”，弹出“创建虚拟私有云”对话框。参考创建虚拟私有云和子网，创建虚拟私有云。 虚拟私有云在Kafka实例创建完成后，不支持修改。
子网	选择已经创建好的或共享的子网。如果您需要创建新子网，单击右侧的“创建子网”，弹出“新建子网”对话框。参考为虚拟私有云创建新的子网，创建子网。 子网在Kafka实例创建完成后，不支持修改。 子网开启IPv6后，Kafka实例支持IPv6功能。
IPv6	子网开启IPv6后，页面才显示此参数。开启IPv6后，客户端可以使用IPv6地址连接实例。 开启IPv6的实例不支持动态开启和关闭SASL_SSL功能。 实例创建成功后，不支持修改IPv6开关。 此功能在“华东二”、“华南-广州”和“华东-上海一”区域上线。
内网IP地址	根据实际需求选择“自动分配”或“手动选择”。 - 自动分配：系统自动分配子网中可用的IP地址。 - 手动选择：在下拉框中勾选指定的IP地址。如果指定的IP地址数量少于代理数量时，系统会自动为剩余代理分配IP地址。 开启IPv6后，控制台不显示此参数。
安全组	选择已经创建好的安全组。 安全组是一组对Kafka实例的访问规则的集合。如果您需要创建新安全组，单击右侧的“创建安全组”，弹出“创建安全组”对话框。参考创建安全组，设置安全组参数。 客户端连接Kafka实例前，请根据不同的连接方式配置对应的安全组规则，否则会连接失败。具体的安全组规则请参考表 5-2。

步骤5 设置实例访问方式。

图 3-5 实例访问方式

访问方式

内网访问

内网接入方式

明文接入

密文接入

实例创建成功后，跨VPC访问安全协议也将默认为 PLAINTEXT 协议，且无法修改。

公网访问

公网接入方式

明文接入

密文接入

公网访问不开启密文接入将会有安全风险!

公网IP地址

请选择

创建弹性IP

请指定弹性IP地址以支持公网访问。应选择 3 个，已选择 0 个。

表 3-12 实例访问方式参数说明

参数名称	子参数	说明
内网访问	接入方式	接入方式分为以下两种： - 明文接入：表示客户端连接Kafka实例时，无需进行SASL认证。 - 密文接入：表示客户端连接Kafka实例时，需要进行SASL认证。 开启“密文接入”后，需要设置kafka安全协议、SSL用户名、密码和SASL PLAIN机制。 内网访问不支持关闭，明文接入和密文接入至少开启一个。 单机实例不支持密文接入。 接入方式的取值决定了跨VPC访问协议的取值，具体如下： - 开启明文接入且关闭密文接入时，跨VPC访问Kafka实例需要使用“PLAINTEXT”协议。 - 开启密文接入且kafka安全协议为“SASL_SSL”时，跨VPC访问Kafka实例需要使用“SASL_SSL”协议。 - 开启密文接入且kafka安全协议为“SASL_PLAINTEXT”时，跨VPC访问Kafka实例需要使用“SASL_PLAINTEXT”协议。 实例创建成功后，跨VPC访问安全协议无法修改。

参数名称	子参数	说明
公网访问	接入方式	接入方式分为以下两种： - 明文接入：表示客户端连接Kafka实例时，无需进行SASL认证。 - 密文接入：表示客户端连接Kafka实例时，需要进行SASL认证。 开启“密文接入”后，需要设置kafka安全协议、SSL用户名、密码和SASL PLAIN机制。 开启公网访问后，明文接入和密文接入至少开启一个，否则无法连接Kafka实例。 单机实例不支持密文接入。
	公网IP地址	根据页面提示信息选择对应数量的公网IP地址。 如果公网IP地址数量不足，请单击“创建弹性IP”跳转到弹性公网IP页面，创建公网IP地址。然后返回购买Kafka实例页面，在“公网IP地址”后单击  ，刷新公网IP地址。 Kafka实例不支持绑定IPv6的弹性公网IP，只支持绑定IPv4的弹性公网IP。

kafka安全协议、SSL用户名、密码和SASL PLAIN机制的参数解释如下：

表 3-13 密文接入参数说明

参数名称	参数值	说明
kafka安全协议	SASL_SSL	采用SASL方式进行认证，数据通过SSL证书进行加密传输，安全性更高。
	SASL_PLAINTEXT	采用SASL方式进行认证，数据通过明文传输，性能更好。 由于明文传输，建议使用SCRAM-SHA-512机制。
SSL用户名	-	客户端用于连接Kafka实例的用户名。 用户名需要符合以下命名规则：由英文字母开头，且只能由英文字母、数字、中划线、下划线组成，长度为4~64个字符。 密文接入成功开启后，用户名不支持修改。

参数名称	参数值	说明
密码	-	客户端用于连接Kafka实例的密码。 密码需要符合以下规则： • 长度为8~32个字符。 • 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=+ [{}];:","<.>?`和空格，并且不能以-开头。 • 不能与用户名或倒序的用户名相同。
SASL认证机制	-	• 未勾选“PLAIN”时，使用SCRAM-SHA-512机制对账号密码进行认证。 • 勾选“PLAIN”后，同时支持SCRAM-SHA-512机制和PLAIN机制，根据实际情况选择其中任何一种配置连接。 密文接入成功开启后，SASL认证机制不支持修改。 什么是SCRAM-SHA-512机制和PLAIN机制？ • SCRAM-SHA-512机制：采用哈希算法对用户名与密码生成凭证，进行身份校验的安全认证机制，比PLAIN机制安全性更高。 • PLAIN机制：一种简单的用户名密码校验机制。

步骤6 设置高级配置。

表 3-14 高级配置参数说明

参数名称	说明
实例名称	名称支持自定义，但需要符合命名规则：长度为4~64个字符，由英文字母开头，只能由英文字母、数字、中划线、下划线组成。
企业项目	该参数针对企业用户使用。 企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理，默认项目为default。
容量阈值策略	设置磁盘使用达到容量阈值后的消息处理策略，容量阈值为95%。 • 自动删除：可以正常生产和消费消息，但是会删除最早的10%的消息，以保证磁盘容量充足。该场景优先保障业务不中断，数据存在丢失的风险。 • 生产受限：无法继续生产消息，但可以继续消费消息。该场景适用于对数据不能丢的业务场景，但是会导致生产业务失败。

参数名称	说明
Smart Connect	选择是否开启Smart Connect。 Smart Connect用于异构系统间的数据同步，开启Smart Connect后，设置Smart Connect任务可以实现Kafka实例和其他云服务之间的数据同步，或者两个Kafka实例之间的数据同步。 开启Smart Connect后，实例会另外创建2个用于Smart Connect的代理。 单机实例不显示此参数。
Kafka自动创建Topic	选择是否开启Kafka自动创建Topic功能。 开启Kafka自动创建Topic功能表示向一个未创建的Topic生产或消费消息时，系统会自动创建此Topic，此Topic的默认参数值如表3-15所示。 ● 单机实例不支持修改配置参数。
标签	标签用于标识云资源，当您拥有相同类型的许多云资源时，可以使用标签按各种维度（例如用途、所有者或环境）对云资源进行分类。 如果您的组织已经设定分布式消息服务Kafka版的相关标签策略，则需按照标签策略规则为Kafka实例添加标签。标签如果不符合标签策略的规则，则可能会导致Kafka实例创建失败，请联系组织管理员了解标签策略详情。 ● 如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键值对。另外，您可以单击“查看预定义标签”，跳转到标签管理服务页面，查看已经预定义的标签，或者创建新的标签。 ● 您也可以直接在“标签键”和“标签值”中设置标签。 当前每个Kafka实例最多支持设置20个不同标签，标签的命名规格，请参考配置Kafka实例标签章节。
描述	设置实例的描述信息，长度为0~1024个字符。

表 3-15 Topic 参数值

参数名称	单机实例默认参数值	集群实例默认参数值	集群实例修改后的参数值
分区数	1	3	5
副本数	1	3	3
老化时间	72小时	72小时	72小时
同步复制	不开启	不开启	不开启
同步落盘	不开启	不开启	不开启
消息时间戳类型	CreateTime	CreateTime	CreateTime

参数名称	单机实例默认参数值	集群实例默认参数值	集群实例修改后的参数值
批处理消息最大值	10485760字节	10485760字节	10485760字节

步骤7 设置实例购买时长。

当选择了“包年/包月”付费模式时，页面才显示“购买时长”参数，您可以根据业务需要选择购买时长。如果勾选“自动续费”，在“包年/包月”资源到期后，自动进行续费。

- 按月购买的实例：自动续费周期为1个月。
- 按年购买的实例：自动续费周期为1年。

步骤8 在页面右侧“配置概要”中，查看已选择的实例配置。

步骤9 确认配置无误后，单击“确认订单”，进入“确认配置”页面。

步骤10 确认实例信息无误且阅读并同意《华为云用户协议》后，如果“计费模式”选择“包年/包月”，单击“去支付”。根据界面提示信息，支付费用。如果“计费模式”选择“按需付费”，单击“提交”。

步骤11 在实例列表页面，查看Kafka实例是否创建成功。

创建实例大约需要3到15分钟，此时实例的“状态”为“创建中”。

- 当实例的“状态”变为“运行中”时，说明实例创建成功。
- 当实例的“状态”变为“创建失败”，请参考[删除Kafka实例](#)，删除创建失败的实例，然后重新创建。如果重新创建仍然失败，请联系客服。

说明

创建失败的实例，不会占用其他资源。

----结束

购买相同配置的 Kafka 实例

若您需购买与当前Kafka实例配置一致的新实例，可通过“购买相同配置”功能，一键复用当前实例配置，快速完成新实例购买，大幅节省配置时间。

步骤1 进入[Kafka实例控制台](#)。

步骤2 在左上角单击，选择区域。

不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。

步骤3 选择目标Kafka实例，单击“操作”列下的“更多 > 购买相同配置”，进入“购买Kafka实例”页面。

步骤4 系统已自动复制目标Kafka实例的参数配置，根据需要调整待购买Kafka实例的参数配置。参数配置的详细内容，请参见[自定义购买Kafka单机/集群实例](#)。

为安全起见，当复制的Kafka实例存在如下场景时，系统不会自动复制相应的参数配置，需要您手动配置：

- 开启密文接入的Kafka实例，需要重新设置SSL用户名和密码。
- 开启公网访问的Kafka实例，需要重新设置公网IP地址。
- 指定内网IP地址的Kafka实例，需要在“内网IP地址”中重新设置为“手动选择”，并指定相应的内网IP地址。
- 目标Kafka实例的名称不会被复制，需要重新设置Kafka实例名称。

步骤5 在页面右侧“配置概要”中，查看已选择的实例配置。

步骤6 确认配置无误后，单击“确认订单”，进入“确认配置”页面。

步骤7 确认实例信息无误且阅读并同意《华为云用户协议》后，如果“计费模式”选择“包年/包月”，单击“去支付”。根据界面提示信息，支付费用。如果“计费模式”选择“按需付费”，单击“提交”。

步骤8 在实例列表页面，查看Kafka实例是否创建成功。

创建实例大约需要3到15分钟，此时实例的“状态”为“创建中”。

- 当实例的“状态”变为“运行中”时，说明实例创建成功。
- 当实例的“状态”变为“创建失败”，请参考[删除Kafka实例](#)，删除创建失败的实例，然后重新创建。如果重新创建仍然失败，请联系客服。

说明

创建失败的实例，不会占用其他资源。

----结束

相关文档

如果您希望通过API购买Kafka实例，请参考[创建Kafka实例](#)。

4 配置 Topic

4.1 创建 Kafka Topic

Topic，即消息主题，用于存储消息，供生产者生产消息以及消费者订阅消息。创建 Kafka 实例成功后，如果未开启“Kafka 自动创建 Topic”功能，则需要手动创建 Topic。如果实例开启了“Kafka 自动创建 Topic”，则该操作为可选。

“Kafka 自动创建 Topic”表示向一个未创建的 Topic 生产或消费消息时，系统会自动创建此 Topic，此 Topic 的默认参数值如表 4-1 所示。

集群实例支持在“实例管理 > 配置参数”中修改“log.retention.hours”（老化时间）、“default.replication.factor”（副本数）或“num.partitions”（分区数）的参数值，单机实例不支持修改配置参数。修改参数值后自动创建的 Topic，Topic 参数值为修改后的值。

例如：“num.partitions”修改为“5”，自动创建的 Topic 参数值如表 4-1 所示。

表 4-1 Topic 参数值

参数名称	单机实例默认值	集群实例默认值	集群实例修改后的值
分区数	1	3	5
副本数	1	3	3
老化时间	72小时	72小时	72小时
同步复制	不开启	不开启	不开启
同步落盘	不开启	不开启	不开启
消息时间戳类型	CreateTime	CreateTime	CreateTime
批处理消息最大值	10485760字节	10485760字节	10485760字节

约束与限制

- Kafka单机/集群实例对Topic的总分区数设置了上限，**当Topic的总分区数达到上限后，用户就无法继续创建Topic**。不同规格配置的Topic总分区数不同，具体请参考[Kafka实例规格](#)。
- 2023年5月17日及以后创建的实例不提供Kafka Manager功能，不支持在Kafka Manager中创建Topic。
- 已开启密文接入的实例，“allow.everyone.if.no.acl.found”为“false”时，初始用户（即首次开启密文接入时设置的用户）可以通过客户端创建Topic，其他用户无法通过客户端创建Topic。
- Topic名称开头包含特殊字符，例如“#”号时，监控数据无法展示。
- 由于Kafka内核限制，无法区分句点和下划线，因此无法创建名称中只有句点和下划线不同的Topic。例如，已创建Topic_1，在新创建Topic.1时，报错如下：Topic 'topic.1' collides with existing topics: topic_1。

创建 Kafka Topic

您可以在Kafka控制台创建Topic，也可以通过Kafka Manager或者客户端创建Topic。

在控制台创建 Topic

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > Topic管理”，单击“创建Topic”，弹出“创建Topic”对话框。

步骤5 填写Topic名称和配置信息，单击“确定”，完成创建Topic。

图 4-1 创建 Topic（集群实例）

创建Topic

Topic 名称

topic-909784299

默认名称规则为：Topic-日期-4位随机数（字母和数字），可自定义修改

分区数 

— | 3 | +

输入范围1-200，Topic创建后，分区数不支持扩容。

副本数 

— | 3 | +

输入范围1-3，建议取3，副本数需要小于等于代理个数。

老化时间（小时） 

— | 72 | +

输入范围1-720，Topic中消息的过期时间，超过时间的消息将被删除，无法被消费。

☐ 同步复制

同步复制指Topic接收到生产消息请求后，将消息复制到所有副本再返回响应。启动同步复制需客户端配置acks=all或-1，否则无效。

☐ 同步落盘

同步落盘指消息即时写入磁盘。启动时，消息即时写入，提高可靠性；关闭时，消息存于内存，不即时写盘。

消息时间戳类型 

LogAppendTime

CreateTime

把代理将消息写入日志的时间作为消息的时间戳。

批处理消息最大值（字节） 

— | 10,485,760 | +

描述（可选）

请输入

0/200 

表 4-2 Topic 参数说明

参数	说明
Topic名称	名称支持自定义，但需要符合命名规则：以英文字母、数字、下划线开头，且只能由英文字母、数字、句点、中划线、下划线组成，长度为3~200个字符。 名称不能与以下内置Topic名称一致： • __consumer_offsets • __transaction_state • __trace • __connect-status • __connect-configs • __connect-offsets • __cloud_event • __site_event • __dms_dial_test • __remote_log_metadata • __remote_log_compact_metadata • __remote_log_metadata_offset 创建Topic后不能修改名称。 由于Kafka内核限制，无法区分句点和下划线，因此无法创建名称中只有句点和下划线不同的Topic。例如，已创建Topic_1，在新创建Topic.1时，报错如下：Topic 'topic.1' collides with existing topics: topic_1。
分区数	Topic的分区数。 如果分区数与消费者数一致，分区数越大消费的并发度越大。 该参数设置为1时，消费消息时会按照先入先出的顺序进行消费。 取值范围：1~200。
副本数	您可以为每个Topic设置副本的数量，Kafka会自动在每个副本上备份数据，当其中一个Broker故障时数据依然是可用的，副本数越大可靠性越高。更多关于副本的介绍，请参见Kafka相关概念。 该参数设置为1时，表示只有一份数据。 取值范围：1~实例的代理数量。 说明 实例节点出现故障的情况下，单副本Topic查询消息时可能会报“内部服务错误”，因此不建议使用单副本Topic。
老化时间（小时）	消息的最长保留时间，消费者必须在此时间结束前消费消息，否则消息将被删除。删除的消息，无法被消费。 取值范围：1~720。

参数	说明
同步复制	表示后端收到生产消息请求并复制给所有副本后，才返回客户端。 开启同步复制后，需要在生产者客户端的配置文件或者生产代码中配置acks=all或者-1，否则无效。 当副本数为1时，不能选择同步复制功能。
同步落盘	表示生产的每条消息都会立即写入磁盘，可靠性更高。关闭同步落盘后，生产的消息存在内存中，不会立即写入磁盘。
消息时间戳类型	定义消息中的时间戳类型，取值如下： - CreateTime：生产者创建消息的时间。 - LogAppendTime：broker将消息写入日志的时间。
批处理消息最大值（字节）	Kafka允许的最大批处理大小，如果在生产客户端配置文件或代码中启用消息压缩，则表示压缩后的最大批处理大小。 如果增加“批处理消息最大值”，且存在消费者版本早于0.10.2，此时消费者的“fetch size”值也必须增加，以便消费者可以获取增加后的批处理大小。 取值范围：0~10485760。
描述	Topic的描述信息，长度为0~200个字符。

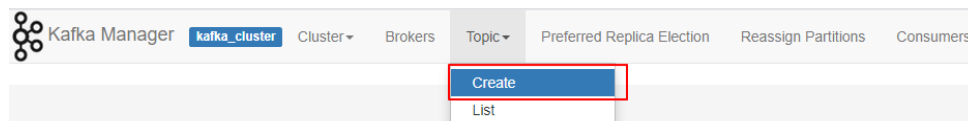
Topic创建完成后，在Topic列表页面查看新创建的Topic。

---结束

在 Kafka Manager 创建 Topic

登录Kafka Manager后，在页面顶部选择“Topic > Create”，然后按照界面参数填写即可。出于性能考虑，建议单个Topic的分区数设置为200以内。

图 4-2 在 Kafka Manager 中创建 Topic



在客户端创建 Topic

Kafka客户端版本为2.2以上时，支持通过kafka-topics.sh创建Topic，以及管理Topic的各类参数。

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令创建Topic。

```
./kafka-topics.sh --create --topic {topic-name} --bootstrap-server {connection-address} --partitions {number-of-partitions} --replication-factor {number-of-replicas}
```

表 4-3 创建 Topic 参数说明

参数名称	说明
topic-name	Topic名称，支持自定义。
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
number-of-partitions	Topic的分区数。 出于性能考虑，建议单个Topic的分区数设置为 200以内 。
number-of-replicas	Topic的副本数。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --create --topic topic-01 --bootstrap-server 192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --partitions 3 --replication-factor 3
Created topic topic-01.
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤创建Topic。
 - a. （可选）如果已经设置了用户名和密码，以及SSL证书配置，请跳过此步骤，执行**b**。否则请执行以下操作。
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加用户名和密码，以及SSL证书配置。
 - b. 在Kafka客户端的“/bin”目录下，通过以下命令创建Topic。
`./kafka-topics.sh --create --topic {topic-name} --bootstrap-server {connection-address} --partitions {number-of-partitions} --replication-factor {number-of-replicas} --command-config ../config/{ssl-user-config.properties}`

表 4-4 创建 Topic 参数说明

参数名称	说明
topic-name	Topic名称，支持自定义。
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
number-of-partitions	Topic的分区数。 出于性能考虑，建议单个Topic的分区数设置为 200以内 。
number-of-replicas	Topic的副本数。
ssl-user-config.properties	配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --create --topic topic-01 --bootstrap-server 192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --partitions 3 --replication-factor 3 --command-config ../config/ssl-user-config.properties
```

```
Created topic topic-01.  
[root@ecs-kafka bin]#
```

相关文档

- 如果您不想手动创建Topic，可以开启自动创建Topic功能，请参考[配置Kafka自动创建Topic](#)。
- 如果您希望通过API创建Topic，请参考[Kafka实例创建Topic](#)。

4.2 设置 Kafka Topic 权限

Kafka实例开启密文接入后，支持对Topic进行ACL（Access Control List）权限管理，您可以在Topic中为不同的用户设置不同的生产消费消息的权限，以达到用户之间的权限隔离。

本章节主要介绍Kafka实例开启密文接入后，如何对Topic进行用户授权。

约束与限制

- 如果“allow.everyone.if.no.acl.found”为“true”，且Topic未授权任何用户，此时所有的用户都可以订阅/发布此Topic。如果Topic已授权某一个或多个用户，此时只有授权的用户可以订阅/发布此Topic，其他未授权的用户不能订阅/发布此Topic。“allow.everyone.if.no.acl.found”在[配置参数](#)中修改。
- 如果“allow.everyone.if.no.acl.found”为“false”，初始用户（即首次开启密文接入时设置的用户）拥有订阅/发布Topic的权限，其他用户在未授权时不能订阅/发布Topic。“allow.everyone.if.no.acl.found”在[配置参数](#)中修改。
- 如果Topic同时设置了“默认用户”权限和单个用户权限，取两者的并集。
- 单机实例不支持设置Topic权限。
- 批量设置Topic权限后，会覆盖Topic之前的权限。
- 修改Topic权限过程中，可能存在权限短暂失效的情况，客户端报“AuthorizationException”错误，建议在客户端设置重试机制，具体方法请参考[Kafka客户端使用建议](#)。

前提条件

- Kafka实例已[开启密文接入](#)。
- 已[创建用户](#)。

查看 Topic 的权限信息

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例名称，进入实例详情页面。

步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。

步骤5 单击Topic名称，进入Topic详情页面。

步骤6 单击“用户权限”，进入“用户权限”页签。

步骤7 查看该Topic的所有授权用户及其对应权限。

图 4-3 Topic 的用户权限



表 4-5 用户权限

参数名称	说明
用户名	拥有发布/订阅Topic权限的用户。
权限	该用户对应的权限。

----结束

设置 Topic 权限

在控制台，您可为SASL用户授予发布/订阅权限，支持为单个Topic授权，也可同时为多个Topic授权。

设置单个 Topic 权限

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 在待设置用户权限的Topic所在行，单击“设置用户权限”，弹出“设置权限”对话框。
- 步骤6 通过以下任意一种方法，进入设置用户权限页面。
 - 在待设置用户权限的Topic所在行，单击“用户权限”，弹出“用户权限”对话框。
 - 单击待设置用户权限的Topic名称，进入Topic详情页面。单击右上角的“设置权限”，弹出“用户权限”对话框。

- 单击待设置用户权限的Topic名称，进入Topic详情页面。单击“用户权限”，进入“用户权限”页签。单击“设置权限”，弹出“用户权限”对话框。

步骤7 为用户设置Topic权限。

- 如果需要为所有用户设置相同的权限，勾选“默认用户”，并设置权限。如下图所示，所有用户都拥有发布此Topic的权限。

图 4-4 为所有用户设置相同的权限

Topic 名称

topic01

分区数

3

老化时间 (小时)

72

副本数

3

同步复制

否

同步落盘

否

☒ 默认用户

发布

勾选默认用户后，将为所有用户统一分配权限，您还可以单独设置用户权限，单独设置的用户权限将和默认用户权限同时生效

- 如果需要为某个用户设置权限，不勾选“默认用户”，在“用户列表”区域，勾选需要订阅/发布/发布+订阅此Topic的用户名。如果用户过多，在搜索框中输入用户名，快速查找用户。在“Topic权限”区域，为用户设置不同的权限。如下图所示，此时只有“test”、“send”和“receive”用户可以订阅/发布此Topic，“send_receive”用户无法订阅/发布此Topic。

图 4-5 为某个用户设置权限

用户列表 选择需要授权的用户

Q 请输入用户名

☒ 用户名

☒ receive

☒ test

☒ send

☐ send_receive

Topic权限

Q 请输入用户名

用户名	权限	操作
send	发布	删除
test	发布+订阅	删除
receive	订阅	删除

同时设置了“默认用户”权限和单个用户权限时，取两者的并集。如下图所示，“test”和“receive”用户可以订阅+发布此Topic，其他用户只能发布此Topic。

图 4-6 为用户设置 Topic 权限

☒ 默认用户

发布

勾选默认用户后，将为所有用户统一分配权限，您还可以单独设置用户权限，单独设置的用户权限将和默认用户权限同时生效

用户列表 选择需要授权的用户

Q 请输入用户名

☒ 用户名

☒ receive

☒ test

☐ send

☐ send_receive

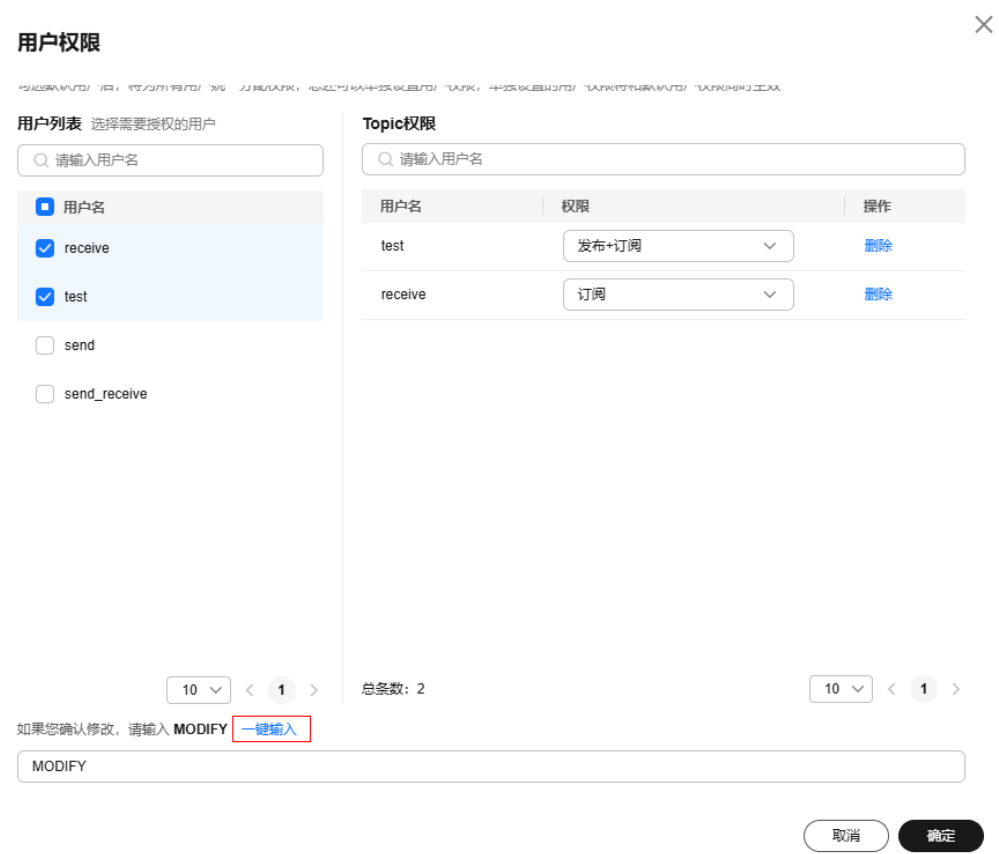
Topic权限

Q 请输入用户名

用户名	权限	操作
test	发布+订阅	删除
receive	订阅	删除

步骤8 在“用户权限”对话框底部，单击“一键输入”，系统自动在输入框中输入“MODIFY”，确认修改权限。然后单击“确定”，完成为用户的授权。

图 4-7 确认修改权限



- 步骤9 验证权限设置是否正确。
1. 单击已设置用户权限的Topic名称，进入Topic详情页面。
 2. 单击“用户权限”，进入“用户权限”页签。
 3. 查看已设置的用户权限。

图 4-8 查看已经授权的用户及其对应的权限



----结束

批量设置 Topic 权限

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 在Topic名称左侧的方框中，勾选待设置用户权限的Topic，单击页面上方的“设置用户权限”，弹出“批量设置权限”对话框。
- 步骤6 为用户批量设置Topic权限。

批量设置权限页面不展示已设置的权限，且设置之后会覆盖之前的权限。例如：“test”用户初始拥有“发布+订阅”Topic01权限，在批量设置Topic权限时，将“test”用户设置为“发布”权限，最终“test”用户只拥有“发布”Topic01权限。

- 如果需要为所有用户设置相同的权限，勾选“默认用户”，并设置权限。如下图所示，所有用户都拥有发布topic01和topic02的权限。

图 4-9 为所有用户设置相同的权限



- 如果需要为某个用户设置权限，不勾选“默认用户”，在“用户列表”区域，勾选待授权的用户名。如果用户过多，在搜索框中输入用户名，快速查找用户。在“Topic权限”区域，为用户设置不同的权限。如下图所示，此时只有“test”、“send”和“receive”用户可以订阅/发布topic01和topic02，“send_receive”用户无法订阅/发布topic01和topic02。

图 4-10 为多个用户设置权限



同时设置了“默认用户”权限和单个用户权限时，取两者的并集。如下图所示，“test”和“receive”用户可以订阅+发布topic01和topic02，其他用户只能发布topic01和topic02。

图 4-11 为用户设置 Topic 权限



步骤7 在“批量设置权限”对话框底部，单击“一键输入”，系统自动在输入框中输入“MODIFY”，确认修改权限。然后单击“确定”，完成为用户的授权。

步骤8 验证权限设置是否正确。

- 1. 单击已设置用户权限的Topic名称，进入Topic详情页面。
- 2. 单击“用户权限”，进入“用户权限”页签。
- 3. 查看已设置的用户权限。

图 4-12 查看已经授权的用户及其对应的权限



----结束

删除单个 Topic 权限

- 步骤1** 登录Kafka控制台。
- 步骤2** 在管理控制台左上角单击📍，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例名称，进入实例详情页面。

- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 通过以下任意一种方法，进入设置用户权限页面。
 - 在待删除用户权限的Topic所在行，单击“用户权限”，弹出“用户权限”对话框。
 - 单击待删除用户权限的Topic名称，进入Topic详情页面。单击右上角的“设置权限”，弹出“用户权限”对话框。
 - 单击待删除用户权限的Topic名称，进入Topic详情页面。单击“用户权限”，进入“用户权限”页签。单击“设置权限”，弹出“用户权限”对话框。
- 步骤6 在“Topic权限”区域，在待删除权限的用户所在行，单击“删除”。
- 步骤7 在“用户权限”对话框底部，单击“一键输入”，系统自动在输入框中输入“MODIFY”，确认修改权限。然后单击“确定”。
- 步骤8 验证权限是否已删除。
 - 单击已删除用户权限的Topic名称，进入Topic详情页面。
 - 单击“用户权限”，进入“用户权限”页签。
 - 已删除的用户不再显示在“用户权限”中，表明权限已成功删除。
- 结束

相关文档

如果您希望通过API设置Topic权限，请参考[设置用户权限](#)。

4.3 Topic 管理

4.3.1 查看 Kafka Topic 详情

在Topic详情页面，您可以查看其基本信息、分区信息、生产者信息和订阅关系。

查看 Topic 详情

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 单击Topic名称，进入Topic详情页面。
- Topic详情页面显示[Topic基本信息](#)、[分区信息](#)、[生产者信息](#)和[订阅关系](#)。
- 结束

Topic 基本信息

图 4-13 Topic 基本信息

Topic 名称	topic-1128625419	关联代理	broker-0,broker-1,broker-2
分区数	3	创建时间	2025/07/03 16:24:30 GMT+08:00

表 4-6 Topic 基本信息

参数名称	说明
Topic名称	该Topic的名称。
关联代理	该Topic已经部署到关联代理中。
分区数	该Topic的分区数。
创建时间	该Topic的创建时间。 注意 创建Topic的时间或方式满足以下任意一种情况时，Topic详情页面不显示创建时间。 2023年7月10日以及之前创建的Topic。 - 自动创建的Topic、在客户端使用命令行创建的Topic、在客户端通过代码创建的Topic、以及使用Kafka Manager创建的Topic。

分区信息

图 4-14 分区信息

分区	生产者	订阅关系	用户权限	监控	消息查询
分区编号	最小偏移量	最早更新时间	最大偏移量	最近更新时间	消息数
0	0	2025/01/02 09:59:12 GMT+...	0	2025/01/02 09:59:12 GMT+...	1
1	0	--	0	--	0
2	0	--	0	--	0

表 4-7 Topic 的分区信息

参数名称	说明
分区编号	该Topic的分区号。
最小偏移量	该分区的最小偏移量。
最早更新时间	该分区下最早一条消息的更新时间。 注意 当分区中的消息都已老化，或尚未生产消息时，“最早更新时间”显示“--”。
最大偏移量	该分区的最大偏移量。
最近更新时间	该分区下最近一条消息的更新时间。 注意 当分区中的消息都已老化，或尚未生产消息时，“最近更新时间”显示“--”。
消息数	该分区中的消息数。

生产者信息

 **注意**

2023年7月10日以及之前创建的Topic，在Topic详情页面不显示“生产者”页签。

图 4-15 生产者信息

分区	生产者	订阅关系	用户权限	监控	消息查询
Broker 地址		生产者地址		加入时间	
192.168.0.178:9092		192.168.0.21:35224		2024/05/17 14:37:06 GMT+08:00	

表 4-8 Topic 的生产者信息

参数名称	说明
Broker地址	生产者连接的Kafka实例Broker地址。
生产者地址	生产者客户端的地址。
加入时间	生产者连接Kafka实例的时间。

订阅关系

 **警告**

如果一个实例中消费组数量超过10000，则不支持查询Topic的订阅关系。

图 4-16 订阅关系

分区	生产者	订阅关系	用户权限	监控	消息查询
消费组名称	消费组状态	Coordinator(ID)	堆积数		
group-test	EMPTY	2	1		

表 4-9 Topic 的订阅关系

参数名称	说明
消费组名称	订阅该Topic的消费组名称。 单击消费组名称，跳转到对应消费组的详情页面，查看消费者列表和消费进度。

参数名称	说明
消费组状态	消费组当前的状态。 - DEAD：消费组内没有任何成员，且没有任何元数据。 - EMPTY：消费组内没有任何成员，存在元数据。 - PREPARING_REBALANCE：准备开启Rebalance。 - COMPLETING_REBALANCE：所有成员加入消费组。 - STABLE：消费组内成员可以正常消费。
Coordinator(ID)	Coordinator组件所在的代理。
堆积数	消费组中剩余可以消费的消息个数。

相关文档

- 如果您希望通过API查询Topic的分区信息，请参考[查询Topic的分区列表](#)。
- 如果您希望通过API查询Topic的生产者信息，请参考[查询Topic的当前生产者列表](#)。

4.3.2 查看 Kafka Topic 日志

Topic日志记录Topic分区选举Leader的详情，包括选举Leader时间、选举Leader的Topic分区和Leader的选举次数等。本章节介绍如何在控制台查看Topic日志。

Topic日志存储在云日志服务（Log Tank Service，简称LTS）中，由LTS提供查询功能。

约束与限制

- 老实例不支持Topic日志功能，具体以控制台为准。
- Topic日志默认存储时间为七天，如果需要延长保留天数，可以在[LTS控制台修改日志组的存储时间](#)。
- Topic日志开启后会在LTS控制台创建对应的日志组、日志流和仪表盘。使用期间按照日志量收费，收费标准请参照[LTS价格详情](#)。
- 频繁产生Topic日志，可能会影响实例性能。
- 不同Kafka实例使用相同的日志组和日志流保存Topic日志时，在每个Kafka实例中都能看到使用同一个日志组和日志流的所有Kafka实例的Topic日志。

前提条件

- 开启Topic日志需要您的账号拥有LTS服务的创建日志组和日志流的权限。
- Kafka实例处于“运行中”时，才能开启/关闭Topic日志。

开启 Topic 日志

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“分析与诊断 > Topic日志”，进入“Topic日志”页面。

步骤5 单击“开启日志”，弹出“开启日志”对话框。

步骤6 单击“确定”，弹出“日志配置”对话框。

步骤7 根据实际情况决定是否开启日志配置，单击“确定”。自动跳转到“后台任务管理”页面，开启日志的任务状态为“成功”时，表示开启Topic日志成功。

- 不开启：LTS后台会自动创建一个日志组以及对应的日志流。
- 开启：选择存储日志文件“topic.log”的日志组和日志流。您还可以单击右侧的“查看日志组”，跳转到LTS控制台，查看或创建日志组和日志流。

----结束

查看 Topic 日志

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“分析与诊断 > Topic日志”，进入“Topic日志”页面。

步骤5 在“日志查询”页签，查看Topic日志。

如果您需要搜索日志，请参考[进入搜索LTS日志页面](#)进行操作。

Topic日志示例如下：

```
{
  "level": "INFO",
  "timestamp": "2024-12-27 17:26:13,361",
  "message": {
    "topicPartition": "topic-0",
    "targetState": "OnlinePartition",
    "leaderAndIsr": "LeaderAndIsr(leader=1, leaderEpoch=3, isr=List(1, 0),
leaderRecoveryState=RECOVERED, partitionEpoch=3)",
    "partitionState": "OnlinePartition",
    "topic": "topic",
    "type": "ELECT_LEADER"
  }
}

{
  "level": "INFO",
  "timestamp": "2024-12-27 17:26:13,491",
  "message": {
    "leader": "1",
    "startOffset": "0",
    "topic": "topic",
    "type": "MAKE_LEADER",
    "topicPartition": "topic-0",
    "epoch": "3"
  }
}
```

参数说明如[表4-10](#)所示。

表 4-10 Topic 日志参数说明

参数名称	说明
level	Topic日志的等级，只有“INFO”一种等级。
timestamp	Topic分区选举Leader或确定Leader的时间。
topicPartition	Topic分区。
targetState	目标状态，取值如下： • NewPartition：表示分区处于新建状态。 • OnlinePartition：表示分区处于正常工作状态。 • OfflinePartition：表示分区处于下线状态。 • NonExistentPartition：表示分区不存在或被删除。
leaderAndlsr	leaderAndlsr请求的信息。
partitionState	分区状态，取值如下： • NewPartition：表示分区处于新建状态。 • OnlinePartition：表示分区处于正常工作状态。 • OfflinePartition：表示分区处于下线状态。 • NonExistentPartition：表示分区不存在或被删除。
topic	Topic名称。
type	Leader所处的阶段，取值如下： • ELECT_LEADER：选举Leader。 • MAKE_LEADER：确定Leader。
leader	Leader所在分区。
startOffset	Leader在对应Epoch上写入第一条消息的Offset。 每个Epoch对应一个startOffset。
epoch	Leader的选举次数，初始值为0。Leader每发生一次选举，Epoch值加一。

----结束

关闭 Topic 日志

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“分析与诊断 > Topic日志”，进入“Topic日志”页面。
- 步骤5 在页面右上角单击“关闭日志”，弹出确认关闭对话框。

步骤6 单击“确定”，自动跳转到“后台任务管理”页面，“状态”为“成功”时，表示关闭Topic日志成功。

关闭Topic日志只是停止Topic日志上报功能，LTS控制台的日志组和日志流仍然保留，还会继续收费，如不需要保留该日志，可以在LTS控制台删除对应的[日志组](#)和[日志流](#)。

----结束

4.3.3 修改 Kafka Topic 配置

本章节指导您在控制台修改如[表4-11](#)所示Kafka Topic的配置。

修改同步复制、同步落盘、消息时间戳类型、批处理消息最大值和描述，不会重启实例。

表 4-11 Kafka Topic 的配置参数

参数名称	说明
分区数	Topic的分区数，修改方法请参见 修改Kafka Topic分区数 。
老化时间	消息的最长保留时间，修改方法请参见 修改Kafka消息老化时间 。
副本数	Topic每个分区的副本数量，修改方法请参见 修改Kafka Topic副本数 。
同步复制	后端收到生产消息请求并复制给所有副本后，才返回客户端。
同步落盘	- 开启：生产的每条消息都会立即写入磁盘，可靠性更高。 - 关闭：生产的消息存在内存中，不会立即写入磁盘。
消息时间戳类型	定义消息中的时间戳类型，取值如下： - CreateTime：生产者创建消息的时间。 - LogAppendTime：broker将消息写入日志的时间。
批处理消息最大值	Kafka允许的最大批处理大小，如果启用消息压缩，则表示压缩后的最大批处理大小。 如果增加“批处理消息最大值”的值，且存在消费者版本早于0.10.2，此时消费者的“fetch size”值也必须增加，以便消费者可以获取增加后的批处理大小。
描述	Topic的描述信息。

约束与限制

- 当副本数为1时，不能开启同步复制功能。
- 开启同步复制后，需要在客户端配置acks=all或者-1，否则无效。
- 批量修改Topic配置时，每次最多修改50个Topic的配置。

修改 Topic 配置

Kafka控制台支持修改单个Topic的配置，也支持批量修改Topic配置。批量修改Topic配置时，无法修改Topic描述信息。

修改单个 Topic 配置

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

步骤5 在待修改配置的Topic所在行，单击“编辑”。

步骤6 在“编辑Topic”对话框中，修改Topic的配置，单击“确定”。

设置成功后，在Topic列表页面，查看修改后的配置。

----结束

批量修改 Topic 配置

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

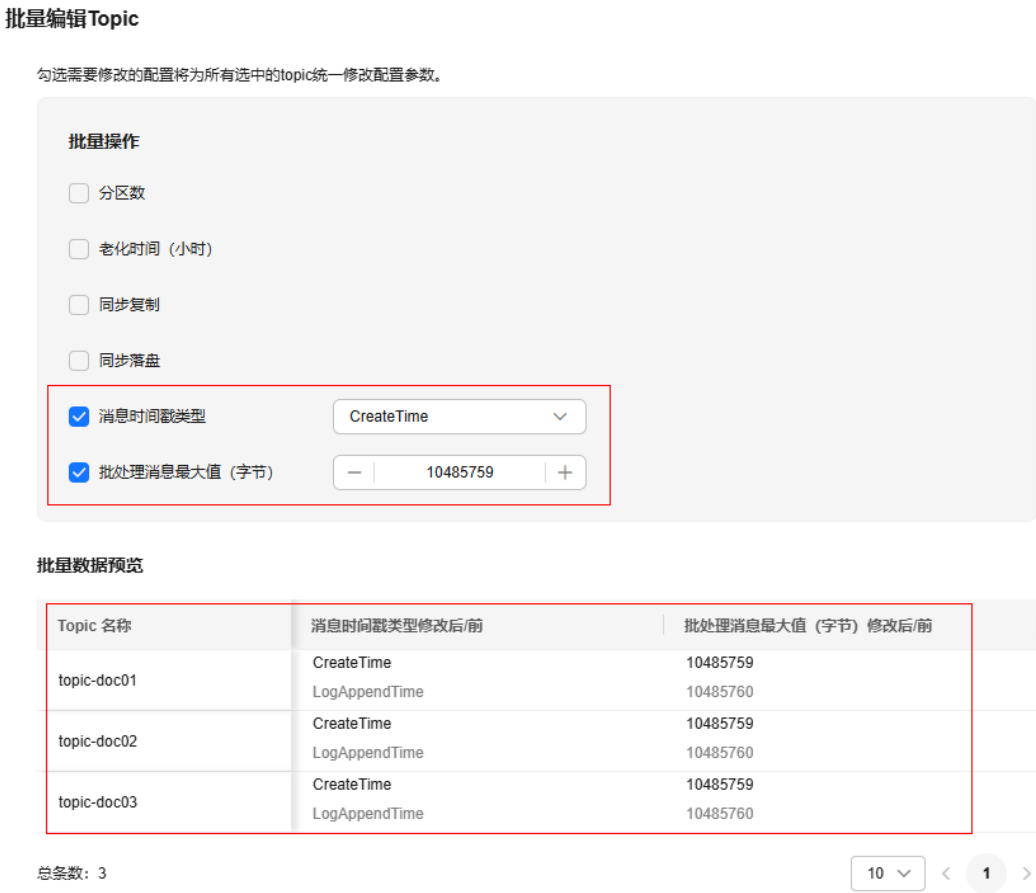
步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

步骤5 勾选待修改配置的Topic，单击信息栏左上侧的“批量编辑Topic”，弹出“批量编辑Topic”对话框。

步骤6 在“批量操作”区域，勾选如[表4-12](#)所示配置。在“批量数据预览”区域，查看修改前和修改后的配置信息。确认无误后，单击“确定”。

图 4-17 批量修改 Topic 配置

总条数：3

10 < 1 >

表 4-12 Topic 配置参数

参数名称	操作说明
同步复制	勾选“同步复制”，并开启或关闭同步复制功能。
同步落盘	勾选“同步落盘”，并开启或关闭同步落盘功能。
消息时间戳类型	勾选“消息时间戳类型”，并在下拉框中选择“CreateTime” / “LogAppendTime”。
批处理消息最大值（字节）	勾选“批处理消息最大值（字节）”，并输入批处理消息最大值。

设置成功后，在Topic列表页面，查看修改后的配置。

----结束

相关文档

如果您希望通过API修改Topic的配置，请参考[修改Kafka实例Topic](#)。

4.3.4 修改 Kafka Topic 分区数

创建Kafka实例Topic成功之后，您可以根据业务需要修改Topic的分区数。**修改分区数，不会重启实例，不会影响业务。**

约束与限制

- 分区数只支持增加，不支持减少。
- 2023年5月17日及以后创建的实例不提供Kafka Manager功能，不支持在Kafka Manager中修改Topic分区数。
- Kafka单机/集群实例对Topic的总分区数设置了上限，**当Topic的总分区数达到上限后，用户就无法继续创建Topic**。不同规格配置的Topic总分区数不同，具体请参考[Kafka实例规格](#)。
- 已开启密文接入的实例，“allow.everyone.if.no.acl.found”为“false”时，初始用户（即首次开启密文接入时设置的用户）可以通过客户端修改Topic分区数，其他用户无法通过客户端修改Topic分区数。
- 批量修改分区数时，每次最多修改50个Topic的分区数。

修改 Kafka Topic 分区数

您可以在Kafka控制台修改分区数，也可以通过Kafka Manager或者客户端修改分区数。

在控制台修改单个 Topic 分区数

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。
- 步骤4** 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5** 在待修改分区数的Topic所在行，单击“编辑”。
- 步骤6** 在“编辑Topic”对话框中，参考[表4-13](#)修改分区数，然后单击“确定”。

表 4-13 分区数参数说明

参数	说明
分区数	输入分区数。分区数只支持增加，不支持减少。 出于性能考虑，Kafka控制台限定单个Topic的分区数上限为200。
新增分区分配	设置新增分区所在代理。 • 自动分配：Kafka服务自动为新增分区分配代理。 • 手动分配：在下拉框中为新增分区分配代理，分配的代理数量不能小于Topic的副本数。

设置成功后，在Topic列表页面，查看修改后的分区数。

----结束

在控制台批量修改 Topic 分区数

- 步骤1 登录Kafka控制台。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 勾选待修改分区数的Topic，单击信息栏左上侧的“批量编辑Topic”，弹出“批量编辑Topic”对话框。
- 步骤6 在“批量操作”区域，勾选“分区数”，并输入分区数。在“批量数据预览”区域，查看修改前和修改后的分区数。确认无误后，单击“确定”。

分区数的取值范围受以下几个条件约束：

- 出于性能考虑，Kafka控制台限定单个Topic的分区数上限为200。
- 分区数不能小于待设置分区数的Topic中最大分区数的值。例如，Topic01的分区数为3，Topic02的分区数为6，批量修改分区数时，设置的分区数不能小于6。

图 4-18 批量修改分区数

批量编辑Topic

勾选需要修改的配置将为所有选中的topic统一修改配置参数。

批量操作

☒ 分区数

-

6

+

☐ 老化时间 (小时)

☐ 同步复制

☐ 同步落盘

☐ 消息时间戳类型

☐ 批处理消息最大值 (字节)

批量数据预览

Topic 名称	分区数修改后/前
topic-doc01	6 2
topic-doc02	6 3
topic-doc03	6 3

总条数: 3

10

<

1

>

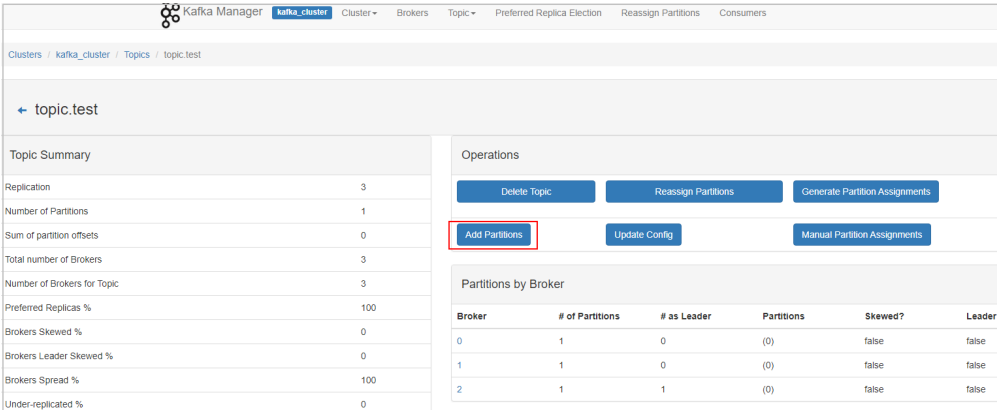
设置成功后，在Topic列表页面，查看修改后的分区数。

----结束

在 Kafka Manager 修改 Topic 分区数

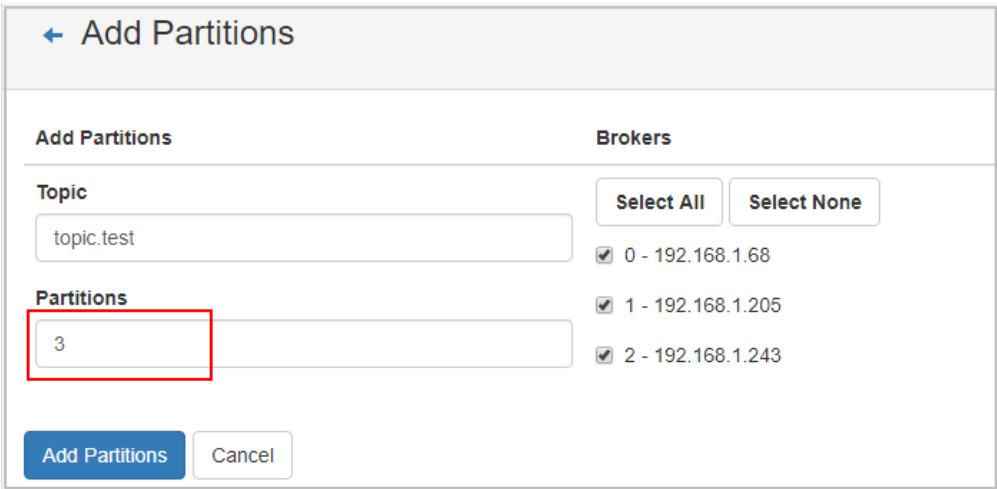
- 步骤1 登录Kafka Manager。
- 步骤2 在Kafka Manager中，单击“Topic > List”，进入Topic列表界面。
- 步骤3 单击Topic名称，进入Topic详情界面。
- 步骤4 单击“Add Partitions”，进入增加分区界面。

图 4-19 Topic 详情界面



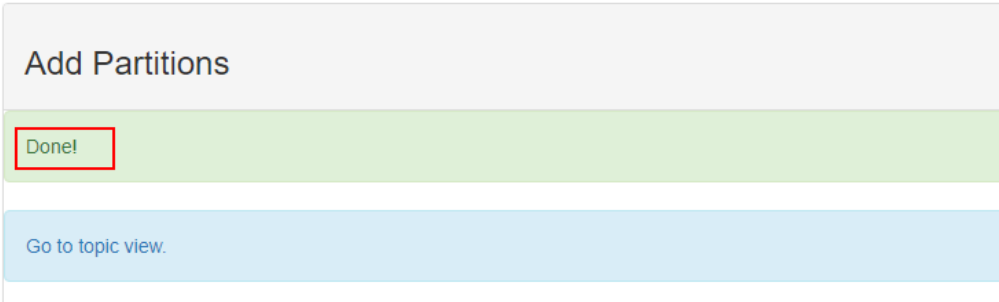
- 步骤5 输入分区数，单击“Add Partitions”。
- 出于性能考虑，建议单个Topic的分区数设置为200以内。

图 4-20 增加分区数量



显示“Done”，表示分区增加成功。

图 4-21 分区增加成功



----结束

在客户端修改 Topic 分区数

Kafka客户端版本为2.2以上时，支持通过kafka-topics.sh修改Topic分区数。

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令修改Topic分区数。
`./kafka-topics.sh --bootstrap-server {connection-address} --topic {topic-name} --alter --partitions {number-of-partitions}`

表 4-14 修改 Topic 分区数参数说明

参数名称	说明
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
topic-name	Topic名称。
number-of-partitions	Topic的分区数。 出于性能考虑，建议单个Topic的分区数设置为200以内。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --bootstrap-server
192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --topic topic-01 --alter --partitions 6
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤修改Topic分区数。
 - （可选）如果已经设置了用户名和密码，以及SSL证书配置，请跳过此步骤，执行b。否则请执行以下操作。
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考步骤3增加用户名和密码，以及SSL证书配置。
 - 在Kafka客户端的“/bin”目录下，通过以下命令修改Topic分区数。
`./kafka-topics.sh --bootstrap-server {connection-address} --topic {topic-name} --alter --partitions {number-of-partitions} --command-config ../config/{ssl-user-config.properties}`

表 4-15 修改 Topic 分区数参数说明

参数名称	说明
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
topic-name	Topic名称。
number-of-partitions	Topic的分区数。 出于性能考虑，建议单个Topic的分区数设置为 200以内 。
ssl-user-config.properties	配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --topic topic-01 --alter --partitions 6 --
command-config ../config/ssl-user-config.properties
[root@ecs-kafka bin]#
```

相关文档

- 如果您希望通过API修改Topic分区数，请参考[修改Kafka实例Topic](#)。
- 如果您希望扩大Topic的总分区数，请参考[如何扩总分区](#)。

4.3.5 修改 Kafka Topic 副本数

创建Kafka实例Topic成功之后，您可以根据业务需要修改Topic的副本数。

约束与限制

- 单机实例不支持修改副本数。
- 分区平衡任务启动后，无法手动停止任务，需要等到任务完成。
- 已设置了一个定时分区平衡任务，在此任务未执行前，无法对本实例内的任何Topic执行其他分区平衡任务。

操作影响

- 对数据量大的Topic进行分区平衡会占用大量的网络和存储带宽，业务可能会出现请求超时或者时延增大，建议在业务低峰期时操作。对Topic进行分区平衡前，根据Kafka实例规格对比当前实例负载情况，评估是否可以分区平衡，建议预留足够的带宽进行分区平衡，CPU使用率在90%以上时，不建议进行分区平衡。Topic的数据量和CPU使用率可以通过监控页面的“队列数据容量”和“CPU使用率”查看，具体步骤请参考[查看Kafka监控数据](#)。
- 带宽限制是指设定Topic进行副本同步的带宽上限，确保不会对该实例上的其他Topic造成流量冲击。但需要注意，带宽限制不会区分是正常的生产消息造成的副本同步还是分区平衡造成的副本同步，如果带宽限制设定过小，可能会影响正常的生产消息，且可能会造成分区平衡一直无法结束。如果分区平衡一直无法结束，请联系客服处理。

- 分区平衡任务启动后，不能删除正在进行分区平衡的Topic，否则会导致分区平衡任务无法结束。
- 分区平衡后Topic的metadata会改变，如果生产者不支持重试机制，会有少量的请求失败，导致部分消息生产失败。
- 数据量大的Topic进行分区平衡的时间会比较长。Topic的数据量可以通过监控页面的“队列数据容量”查看，具体步骤请参考[查看Kafka监控数据](#)。在不影响业务的前提下，建议适当[调小Topic老化时间](#)并等待消息老化，减少迁移数据，加快迁移速度。分区平衡任务结束后可重新调整为初始值。

前提条件

确保分区平衡的目标Broker磁盘容量充足，在[磁盘存储统计](#)中查看每个Broker的可用磁盘容量。如果目标Broker磁盘剩余容量接近分区平衡迁移到该Broker上的数据量，为了给目标Broker上的消息生产预留存储空间，应先进行[磁盘扩容](#)，再进行分区平衡。

修改 Topic 副本数

Kafka控制台提供两种修改副本数的方法：

- 使用自动平衡修改副本数：支持批量修改Topic的副本数。
- 使用手动平衡修改副本数：每次只能修改一个Topic的副本数。

使用自动平衡修改副本数

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。
- 步骤4** 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5** 通过以下任意一种方法，进入“自动平衡”页面。
- 勾选Topic名称左侧的方框，可选一个或多个，单击信息栏左上侧的“一键平衡 > 自动平衡”。
 - 在待修改副本数的Topic所在行，单击“更多 > 一键平衡 > 自动平衡”。
- 步骤6** 修改副本数。

表 4-16 自动平衡参数说明

参数名称	说明
Broker名称	勾选目标Broker的名称，Topic分区的副本即将迁移至目标Broker上。
副本数	输入Topic的副本数，副本数须小于等于Broker的数量。

参数名称	说明
带宽限制	输入带宽大小，默认值为“-1”，表示不限制带宽。 如果实例负载较低（例如实例规格为300MB/s，当前实例负载为30MB/s），建议不设置带宽限制。如果需要设置带宽限制，建议该参数值不小于待分区平衡Topic的总生产带宽 * 待分区平衡Topic的最大副本数。 带宽限制值的估算方法，请参考 带宽限制值估算方法 。
执行时间	选择分区平衡任务执行的时间。 - 立即执行：表示立即执行分区平衡任务。 - 定时执行：表示在指定的时间执行分区平衡任务。

步骤7 （可选）单击“一键测算”，在“预估耗时”中，显示执行自动平衡大概需要的时间。

一键测算功能对Kafka实例性能没有影响。

步骤8 单击“确定”。

定时任务和非定时任务查看分区平衡是否完成的方法不同，具体如下：

表 4-17 查看分区平衡结果

任务类型	分区平衡结果
非定时任务	在Topic列表页面左上方单击“查看变更任务”，进入“后台任务管理 > 即时任务”页面，当任务的“状态”为“成功”时，表示分区平衡完成，即修改副本数成功。
定时任务	- 系统自动跳转到“后台任务管理 > 定时任务”页面，此页面的状态仅表示定时任务是否开始执行，并非表示任务是否执行成功。 - 当“状态”为“待执行”时，表示定时分区平衡任务未执行。 - 当“状态”为“成功”时，表示定时分区平衡任务开始执行。 - 当“状态”为“取消”时，表示已取消定时分区平衡任务。 - 单击“即时任务”，进入“即时任务”页签，查看任务的状态。当任务的“状态”为“成功”时，表示分区平衡完成，即修改副本数成功。

----结束

使用手动平衡修改副本数

步骤1 登录[Kafka控制台](#)。

- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 通过以下任意一种方法，进入“手动平衡”页面。
 - 勾选Topic名称左侧的方框，手动平衡每次只能勾选一个Topic，单击信息栏左上侧的“一键平衡 > 手动平衡”。
 - 在待修改副本数的Topic所在行，单击“更多 > 一键平衡 > 手动平衡”。
- 步骤6 修改副本数。
 - 在“手动平衡”对话框右上角，单击“减少副本”/“添加副本”，为Topic的每个分区减少/增加副本数。
 - 在待修改分区平衡的副本名称下，单击Broker名称或，选择目标Broker的名称，副本即将迁移至目标Broker上。同一分区下的不同副本需要分配在不同的Broker上。
 - 在“带宽限制”中，输入带宽大小，默认值为“-1”，表示不限制带宽。如果实例负载较低（例如实例规格为300MB/s，当前实例负载为30MB/s），建议不设置带宽限制。如果需要设置带宽限制，建议该参数值不小于待分区平衡Topic的总生产带宽 * 待分区平衡Topic的最大副本数。带宽限制值的估算方法，请参考[带宽限制值估算方法](#)。
 - 在“执行时间”中，选择分区平衡任务执行的时间。“立即执行”表示立即执行分区平衡任务。“定时执行”表示在指定的时间执行分区平衡任务。
- 步骤7 （可选）单击“一键测算”，在“预估耗时”中，显示执行手动平衡大概需要的时间。

一键测算功能对Kafka实例性能没有影响。
- 步骤8 单击“确定”。
- 定时任务和非定时任务查看分区平衡是否完成的方法不同，具体如下：

表 4-18 查看分区平衡结果

任务类型	分区平衡结果
非定时任务	在Topic列表页面左上方单击“查看变更任务”，进入“后台任务管理 > 即时任务”页面，当任务的“状态”为“成功”时，表示分区平衡完成，即修改副本数成功。

任务类型	分区平衡结果
定时任务	1. 系统自动跳转到“后台任务管理 > 定时任务”页面，此页面的状态仅表示定时任务是否开始执行，并非表示任务是否执行成功。 - 当“状态”为“待执行”时，表示定时分区平衡任务未执行。 - 当“状态”为“成功”时，表示定时分区平衡任务开始执行。 - 当“状态”为“取消”时，表示已取消定时分区平衡任务。 2. 单击“即时任务”，进入“即时任务”页签，查看任务的状态。当任务的“状态”为“成功”时，表示分区平衡完成，即修改副本数成功。

----结束

相关文档

如果您希望通过API修改副本数，请参考[修改Kafka实例Topic分区的副本](#)。

4.3.6 导出 Kafka Topic 列表

本章节指导您在控制台导出Topic列表，支持批量导出。

前提条件

已[创建Topic](#)。

导出 Topic 列表

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 通过以下任意一种方法，导出Topic列表。
 - 勾选待导出的Topic，单击“导出 > 导出已选中数据到XLSX”，导出指定Topic列表。
 - 单击“导出 > 导出全部数据到XLSX”，导出全部Topic列表。

Topic列表中包含如下信息：Topic名称、分区数、副本数、老化时间、是否同步复制或同步落盘、消息时间戳类型、批处理消息最大值和描述信息。

----结束

4.3.7 修改 Kafka 分区平衡

分区平衡是指将分区的副本重新分配到不同的代理上，解决代理负载不均衡问题。

需要进行分区平衡的场景如下：

- 实例扩容代理个数后，新创建的Topic才会分布在新代理上，原有Topic还分布在原有代理上，造成分区分布不均衡。通过修改分区平衡，将原有Topic分区的副本迁移到新代理上。
- 将高负载代理上的Leader分区切换为Follower。
- 在修改分区平衡过程中，可以配置增加/减少Topic的副本数。

约束与限制

- 单机实例不支持修改分区平衡。
- 分区平衡任务启动后，无法修改Topic的分区数。
- 分区平衡任务启动后，无法手动停止任务，需要等到任务完成。
- 已设置了一个定时分区平衡任务，在此任务未执行前，无法对本实例内的任何Topic执行其他分区平衡任务。

操作影响

- 对数据量大的Topic进行分区平衡会占用大量的网络和存储带宽，业务可能会出现请求超时或者时延增大，建议在业务低峰期时操作。对Topic进行分区平衡前，根据Kafka实例规格对比当前实例负载情况，评估是否可以分区平衡，建议预留足够的带宽进行分区平衡，CPU使用率在90%以上时，不建议进行分区平衡。Topic的数据量和CPU使用率可以通过监控页面的“队列数据容量”和“CPU使用率”查看，具体步骤请参考[查看Kafka监控数据](#)。
- 带宽限制是指设定Topic进行副本同步的带宽上限，确保不会对该实例上的其他Topic造成流量冲击。但需要注意，带宽限制不会区分是正常的生产消息造成的副本同步还是分区平衡造成的副本同步，如果带宽限制设定过小，可能会影响正常的生产消息，且可能会造成分区平衡一直无法结束。如果分区平衡一直无法结束，请联系客服处理。
- 分区平衡任务启动后，不能删除正在进行分区平衡的Topic，否则会导致分区平衡任务无法结束。
- 分区平衡后Topic的metadata会改变，如果生产者不支持重试机制，会有少量的请求失败，导致部分消息生产失败。
- 数据量大的Topic进行分区平衡的时间会比较长。Topic的数据量可以通过监控页面的“队列数据容量”查看，具体步骤请参考[查看Kafka监控数据](#)。在不影响业务的前提下，建议适当[调小Topic老化时间](#)并等待消息老化，减少迁移数据，加快迁移速度。分区平衡任务结束后可重新调整为初始值。

前提条件

确保分区平衡的目标Broker磁盘容量充足，在[磁盘存储统计](#)中查看每个Broker的可用磁盘容量。如果目标Broker磁盘剩余容量接近分区平衡迁移到该Broker上的数据量，为了给目标Broker上的消息生产预留存储空间，应先进行[磁盘扩容](#)，再进行分区平衡。

修改 Kafka 分区平衡

Kafka控制台提供两种分区平衡的方法：

- 自动平衡：支持批量修改Topic的分区平衡。

- 手动平衡：每次只能修改一个Topic的分区平衡。

自动平衡

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 通过以下任意一种方法，修改分区平衡。
 - 勾选Topic名称左侧的方框，可选一个或多个，单击信息栏左上侧的“一键平衡 > 自动平衡”。
 - 在待修改分区平衡的Topic所在行，单击“更多 > 一键平衡 > 自动平衡”。
- 步骤6 设置自动平衡参数。

表 4-19 自动平衡参数说明

参数名称	说明
Broker名称	勾选目标Broker的名称，Topic分区的副本即将迁移至目标Broker上。
副本数	输入Topic需要自动平衡的副本数，副本数须小于等于Broker的数量。
带宽限制	输入带宽大小，默认值为“-1”，表示不限制带宽。 如果实例负载较低（例如实例规格为300MB/s，当前实例负载为30MB/s），建议不设置带宽限制。如果需要设置带宽限制，建议该参数值不小于待分区平衡Topic的总生产带宽 * 待分区平衡Topic的最大副本数。 带宽限制值的估算方法，请参考 带宽限制值估算方法 。
执行时间	选择分区平衡任务执行的时间。 ● 立即执行：表示立即执行分区平衡任务。 ● 定时执行：表示在指定的时间执行分区平衡任务。

图 4-22 设置自动平衡参数



- 步骤7** （可选）单击“一键测算”，在“预估耗时”中，显示执行自动平衡大概需要的时间。
- 一键测算功能对Kafka实例性能没有影响。
- 步骤8** 单击“确定”。
- 定时任务和非定时任务查看分区平衡是否完成的方法不同，具体如下：

表 4-20 查看分区平衡结果

任务类型	分区平衡结果
非定时任务	在Topic列表页面左上方单击“查看变更任务”，进入“后台任务管理 > 即时任务”页面，当任务的“状态”为“成功”时，表示分区平衡完成。

任务类型	分区平衡结果
定时任务	1. 系统自动跳转到“后台任务管理 > 定时任务”页面，此页面的状态仅表示定时任务是否开始执行，并非表示任务是否执行成功。 - 当“状态”为“待执行”时，表示定时分区平衡任务未执行。 - 当“状态”为“成功”时，表示定时分区平衡任务开始执行。 - 当“状态”为“取消”时，表示已取消定时分区平衡任务。 2. 单击“即时任务”，进入“即时任务”页签，查看任务的状态。当任务的“状态”为“成功”时，表示分区平衡完成。

图 4-23 后台任务管理页面



说明

- 分区平衡任务启动后，不能删除正在进行分区平衡的Topic，否则会导致分区平衡任务无法结束。
- 分区平衡任务启动后，无法修改Topic的分区数。
- 分区平衡任务启动后，无法手动停止任务，需要等到任务完成。
- 已设置了一个定时分区平衡任务，在此任务未执行前，无法对本实例内的任何Topic执行其他分区平衡任务。

----结束

手动平衡

- 步骤1 登录Kafka控制台。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5 通过以下任意一种方法，修改分区平衡。
 - 勾选Topic名称左侧的方框，手动平衡每次只能勾选一个Topic，单击信息栏左上侧的“一键平衡 > 手动平衡”。
 - 在待修改分区平衡的Topic所在行，单击“更多 > 一键平衡 > 手动平衡”。

- 步骤6** 设置手动平衡参数。
- 在“手动平衡”对话框右上角，单击“减少副本” / “添加副本”，为Topic的每个分区减少/增加副本数。
 - 在待修改分区平衡的副本名称下，单击Broker名称或，选择目标Broker的名称，副本即将迁移至目标Broker上。同一分区下的不同副本需要分配在不同的Broker上。
 - 在“带宽限制”中，输入带宽大小，默认值为“-1”，表示不限制带宽。如果实例负载较低（例如实例规格为300MB/s，当前实例负载为30MB/s），建议不设置带宽限制。如果需要设置带宽限制，建议该参数值不小于待分区平衡Topic的总生产带宽 * 待分区平衡Topic的最大副本数。带宽限制值的估算方法，请参考[带宽限制值估算方法](#)。
 - 在“执行时间”中，选择分区平衡任务执行的时间。“立即执行”表示立即执行分区平衡任务。“定时执行”表示在指定的时间执行分区平衡任务。

图 4-24 设置手动平衡参数

手动平衡 ✕

每个分区下的副本1为优先leader副本

test01 减少副本 添加副本

分区 0

副本 1
Broker 0 

副本 2
Broker 2 

副本 3
Broker 1 

分区 1

副本 1

副本 2

副本 3

带宽限制

-

-1

+

MB/s

(取值范围: 1~300 MB/s; 默认值为-1, 表示不限流)

分区平衡过程中可能会对业务造成影响

[查看详情](#) 

执行时间

定时执行 

2024/08/21 16:32:40 

预估耗时

配置完成后点击下方一键测算

取消

一键测算

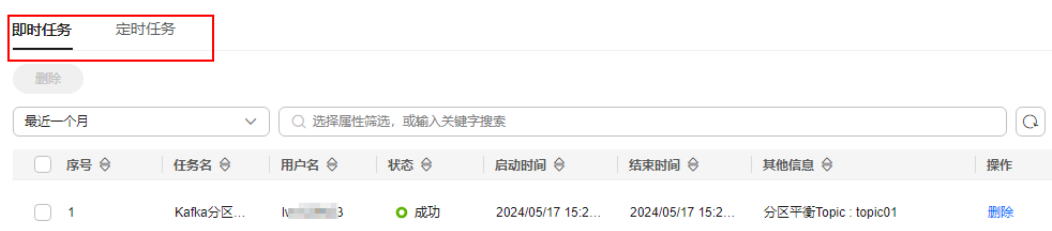
确定

- 步骤7** （可选）单击“一键测算”，在“预估耗时”中，显示执行手动平衡大概需要的时间。
- 一键测算功能对Kafka实例性能没有影响。
- 步骤8** 单击“确定”。
- 定时任务和非定时任务查看分区平衡是否完成的方法不同，具体如下：

表 4-21 查看分区平衡结果

任务类型	分区平衡结果
非定时任务	在Topic列表页面左上方单击“查看变更任务”，进入“后台任务管理 > 即时任务”页面，当任务的“状态”为“成功”时，表示分区平衡完成。
定时任务	1. 系统自动跳转到“后台任务管理 > 定时任务”页面，此页面的状态仅表示定时任务是否开始执行，并非表示任务是否执行成功。 - 当“状态”为“待执行”时，表示定时分区平衡任务未执行。 - 当“状态”为“成功”时，表示定时分区平衡任务开始执行。 - 当“状态”为“取消”时，表示已取消定时分区平衡任务。 2. 单击“即时任务”，进入“即时任务”页签，查看任务的状态。当任务的“状态”为“成功”时，表示分区平衡完成。

图 4-25 后台任务管理页面



说明

- 分区平衡任务启动后，不能删除正在进行分区平衡的Topic，否则会导致分区平衡任务无法结束。
- 分区平衡任务启动后，无法修改Topic的分区数。
- 分区平衡任务启动后，无法手动停止任务，需要等到任务完成。
- 已设置了一个定时分区平衡任务，在此任务未执行前，无法对本实例内的任何Topic执行其他分区平衡任务。

----结束

修改定时分区平衡任务

步骤1 在“实例管理 > 后台任务管理”页面的“定时任务”页签中，单击页面左上角下拉框，选择时间段，在搜索对话框中输入待修改定时分区平衡任务的Topic名称，按“Enter”，实现快速查找定时分区平衡任务。

图 4-26 查找定时分区平衡任务



- 步骤2** 在待修改的定时分区平衡任务后，单击“修改”。
- 步骤3** 在弹出的“修改定时任务”对话框中，您可以修改定时分区平衡任务的时间，还可以取消定时分区平衡任务，具体操作如下。
- 修改定时分区平衡任务的时间：修改时间，单击“确定”。
 - 取消定时分区平衡任务：选择“取消”，如图4-27所示，单击“确定”。

图 4-27 取消定时分区平衡任务



----结束

带宽限制值估算方法

带宽限制值受分区平衡任务执行时间、分区副本Leader/Follower分布情况以及消息生产速率等因素影响，具体分析如下。

- 带宽限制值作用范围为整个Broker，对该Broker内所有副本同步的分区进行带宽限流。
- 带宽限制会将分区平衡后新增的副本视为Follower副本限流，分区平衡前的Leader副本视为Leader副本限流，Leader副本的限流和Follower副本限流分开计算。
- 带宽限制不会区分是正常的生产消息造成的副本同步还是分区平衡造成的副本同步，因此两者的流量都会被限流统计。

假设分区平衡任务需要在200s内完成，每个副本的数据量为100MB，在以下几种场景中，估算带宽限制值。

场景一： Topic1有2分区2副本，Topic2有1分区1副本，所有Leader副本在同一个Broker上，如表4-22所示，Topic1和Topic2分别需要新增1个副本，如表4-23所示。

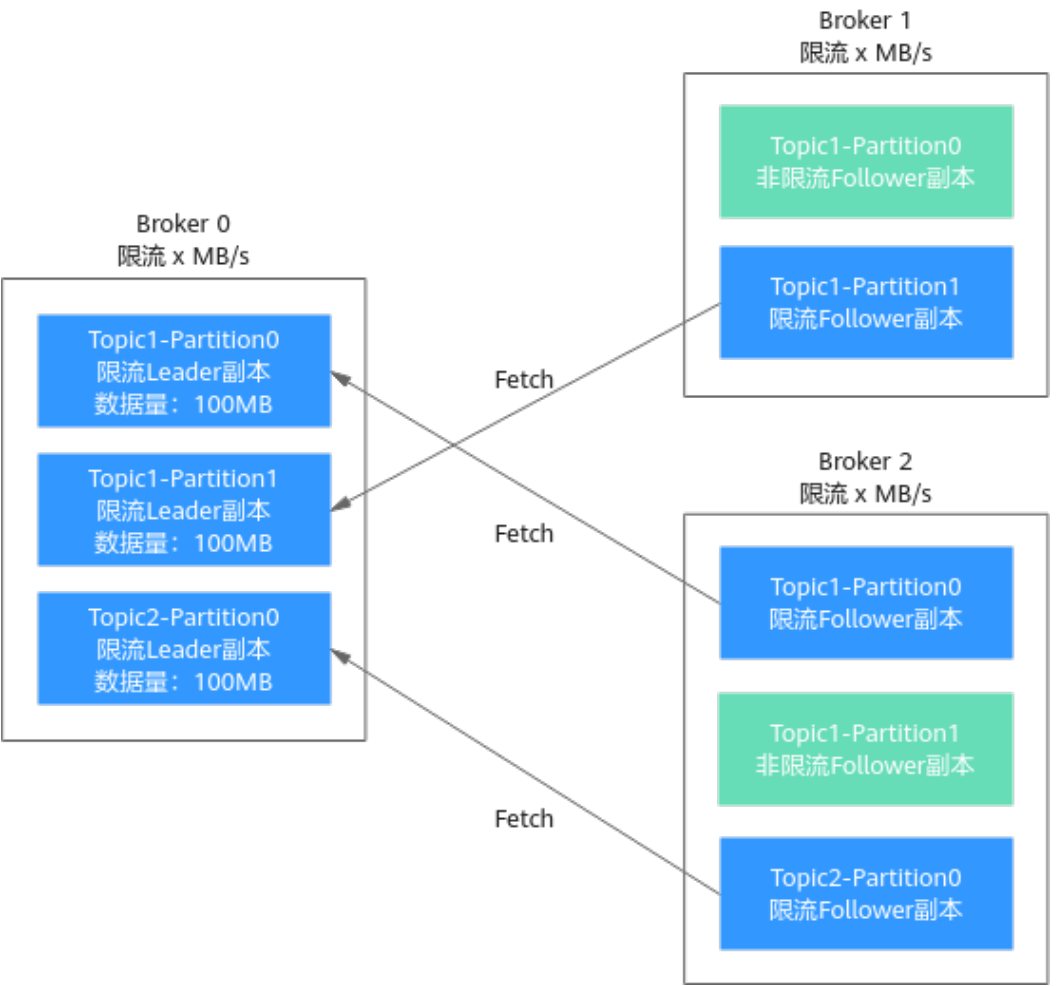
表 4-22 分区平衡前 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0, 1
Topic1	1	0	0, 2
Topic2	0	0	0

表 4-23 分区平衡后 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0, 1, 2
Topic1	1	0	0, 1, 2
Topic2	0	0	0, 2

图 4-28 场景一分区平衡图



如图4-28所示，有3个副本需要从Broker 0拉取数据，Broker 0中每个副本的数据量为100MB，Broker 0中只有Leader副本，Broker 1和Broker 2中只有Follower副本，由此得出以下数据：

- Broker 0在200s内完成分区平衡所需的带宽限制值=（ 100+100+100 ） / 200=1.5MB/s
- Broker 1在200s内完成分区平衡所需的带宽限制值=100/200=0.5MB/s
- Broker 2在200s内完成分区平衡所需的带宽限制值=（ 100+100 ） /200=1MB/s

综上所述，若想要在200s内完成分区平衡任务，带宽限制值应设为大于等于1.5MB/s。由于控制台的带宽限制只能是整数，因此带宽限制值应设为大于等于2MB/s。

场景二：Topic1有2分区1副本，Topic2有2分区1副本，Leader副本分布在不同Broker上，如表4-24所示，Topic1和Topic2分别需要新增1个副本，如表4-25所示。

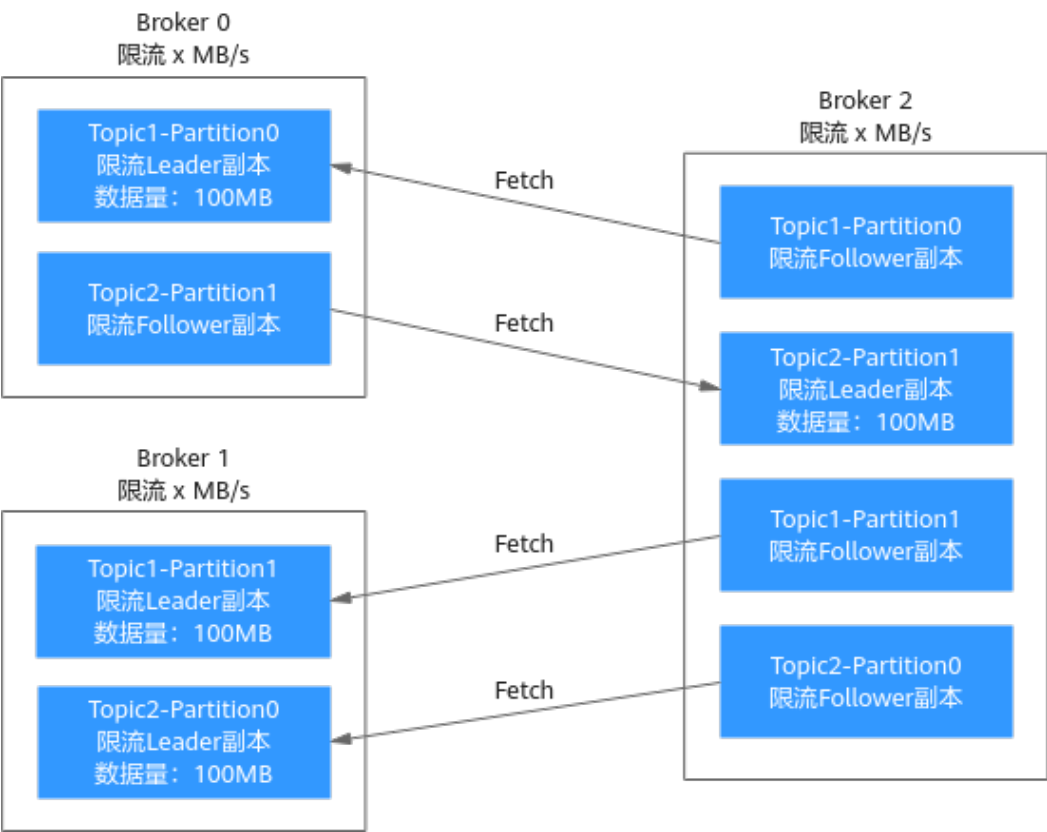
表 4-24 分区平衡前 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0
Topic1	1	1	1
Topic2	0	1	1
Topic2	1	2	2

表 4-25 分区平衡后 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0, 2
Topic1	1	1	1, 2
Topic2	0	1	1, 2
Topic2	1	2	2, 0

图 4-29 场景二分区平衡图



如图4-29所示，Broker 1中只有Leader副本，Broker 0和Broker 2中存在Leader副本和Follower副本，由于Leader副本的限流和Follower副本限流分开计算，Broker 0和Broker 2需要分别计算Leader副本的限流和Follower副本限流。由此得出以下数据：

- Broker 0作为Leader副本在200s内完成分区平衡所需的带宽限制值
=100/200=0.5MB/s
- Broker 0作为Follower副本在200s内完成分区平衡所需的带宽限制值
=100/200=0.5MB/s
- Broker 1在200s内完成分区平衡所需的带宽限制值= (100+100) /200=1MB/s
- Broker 2作为Leader副本在200s内完成分区平衡所需的带宽限制值
=100/200=0.5MB/s
- Broker 2作为Follower副本在200s内完成分区平衡所需的带宽限制值=
(100+100+100) /200=1.5MB/s

综上所述，若想要在200s内完成分区平衡任务，带宽限制值应设为大于等于1.5MB/s。由于控制台的带宽限制只能是整数，因此带宽限制值应设为大于等于2MB/s。

场景三：Topic1有1分区2副本，Topic2有1分区2副本，所有Leader副本在同一个Broker上，如表4-26所示，Topic1需要新增1个副本，Topic1上有生产消息造成的副本同步，如表4-27所示。

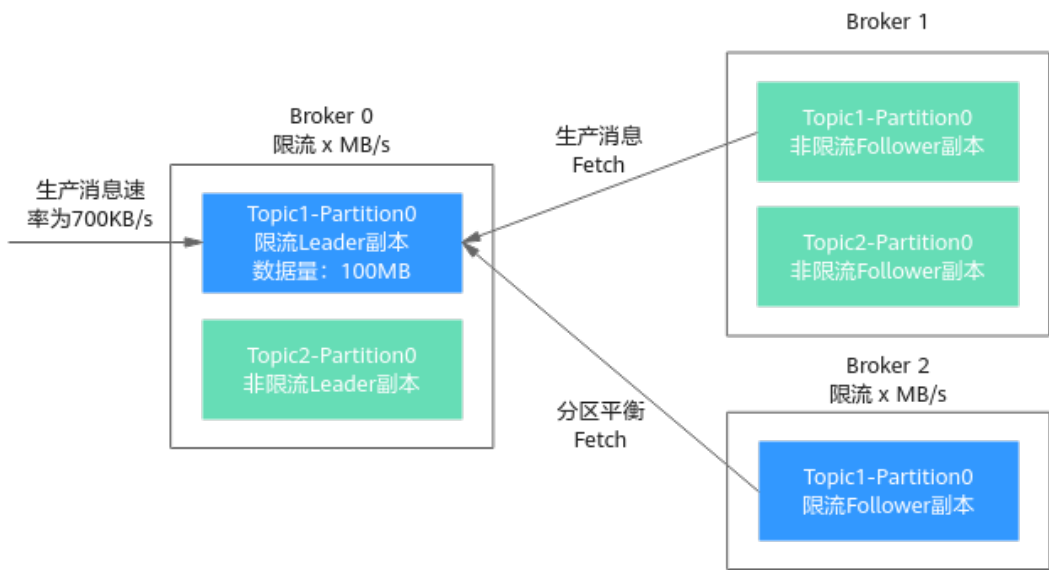
表 4-26 分区平衡前 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0, 1
Topic2	0	0	0, 1

表 4-27 分区平衡后 Topic 分区的副本分布

Topic名称	分区名称	Leader副本所在Broker	Follower副本所在Broker
Topic1	0	0	0, 1, 2
Topic2	0	0	0, 1

图 4-30 场景三分区平衡图



如图4-30所示，有1个副本由于分区平衡需要从Broker 0拉取数据，另1个副本由于生产消息需要从Broker 0拉取数据，带宽限制不会区分是正常的生产消息造成的副本同步还是分区平衡造成的副本同步，因此两者的流量都会被限流统计。由此得出以下数据：

- Broker 0在200s内完成分区平衡所需的带宽限制值= (100MB+700KB/s*200s) / 200s+700KB/s=1.9MB/s
- Broker 2在200s内完成分区平衡所需的带宽限制值=100/200=0.5MB/s

综上所述，若想要在200s内完成分区平衡任务，带宽限制值应设为大于等于1.9MB/s。由于控制台的带宽限制只能是整数，因此带宽限制值应设为大于等于2MB/s。

相关文档

- 如果您希望通过API修改分区平衡，请参考[修改Kafka实例Topic分区的副本](#)。
- 如果您想要了解不同分区间业务数据不均衡的原因和处理措施，请参考[业务数据不均衡处理建议](#)。

4.3.8 配置 Kafka 自动创建 Topic

“Kafka自动创建Topic”表示向一个未创建的Topic生产或消费消息时，系统会自动创建此Topic，此Topic的默认参数值如[表4-28](#)所示。

集群实例支持在“实例管理 > 配置参数”中修改“log.retention.hours”（老化时间）、“default.replication.factor”（副本数）或“num.partitions”（分区数）的参数值，单机实例不支持修改配置参数。修改参数值后自动创建的Topic，Topic参数值为修改后的值。

例如：“num.partitions”修改为“5”，自动创建的Topic参数值如[表4-28](#)所示。

表 4-28 Topic 参数值

参数名称	单机实例默认值	集群实例默认值	集群实例修改后的值
分区数	1	3	5
副本数	1	3	3
老化时间	72小时	72小时	72小时
同步复制	不开启	不开启	不开启
同步落盘	不开启	不开启	不开启
消息时间戳类型	CreateTime	CreateTime	CreateTime
批处理消息最大值	10485760字节	10485760字节	10485760字节

约束与限制

开启/关闭“Kafka自动创建Topic”可能会导致实例重启，具体以控制台显示为准。

配置 Kafka 自动创建 Topic

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入“概览”页面。
- 步骤4 在“实例信息”区域的“Kafka自动创建Topic”后，单击或，弹出“确认”对话框。

步骤5 单击“确定”，跳转到“后台任务管理”页面。任务状态为“成功”时，表示开启/关闭Kafka自动创建Topic成功。

----结束

相关文档

- 如果您希望通过API开启/关闭自动创建Topic功能，请参考[开启或关闭实例自动创建Topic功能](#)。
- 如果您希望手动创建Topic，请参考[创建Kafka Topic](#)。

4.3.9 删除 Kafka Topic

本章节指导您删除Topic，请根据实际情况选择以下任意一种方式：

- [删除Kafka Topic（控制台）](#)
- [删除Kafka Topic（Kafka客户端）](#)

约束与限制

- 删除Topic会导致Topic中的数据也被删除，并且无法找回，同时会影响使用该Topic的相关业务。
- 已开启密文接入的实例，“allow.everyone.if.no.acl.found”设置为“false”时，初始用户（即首次开启密文接入时设置的用户）可以通过客户端删除Topic，其他用户无法通过客户端删除Topic。

前提条件

只有运行中的实例才可以删除Topic。

删除 Kafka Topic

您可以通过控制台删除Topic，也可以通过客户端删除Topic。

删除 Kafka Topic（控制台）

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

步骤5 通过以下任意一种方法，删除Topic。

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

步骤6 在“删除Topic”对话框中，单击“确定”。

已删除的Topic不再显示在Topic列表中，表明Topic已成功删除。

----结束

删除 Kafka Topic（Kafka 客户端）

Kafka客户端版本为2.2以上时，支持通过**kafka-topics.sh**删除Topic。

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令删除Topic。
`./kafka-topics.sh --bootstrap-server {connection-address} --delete --topic {topic-name}`

表 4-29 删除 Topic 参数说明

参数名称	说明
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
topic-name	Topic名称。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --bootstrap-server
192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --delete --topic topic-01
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤删除Topic。
 - （可选）如果已经设置了SSL证书配置，请跳过此步骤，执行b。否则请执行以下操作。
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加SSL证书配置。
 - 在Kafka客户端的“/bin”目录下，通过以下命令删除Topic。
`./kafka-topics.sh --bootstrap-server {connection-address} --delete --topic {topic-name} --command-config ../config/{ssl-user-config.properties}`

表 4-30 删除 Topic 参数说明

参数名称	说明
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
topic-name	Topic名称。
ssl-user-config.properties	配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-topics.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --delete --topic topic-01 --command-
config ../config/ssl-user-config.properties
[root@ecs-kafka bin]#
```

相关文档

- 如果您希望通过API删除Topic，请参考**Kafka实例批量删除Topic**。

- 如果您删除Topic后，发现Topic依然存在，请参考[删除Topic失败](#)进行处理。

5 连接实例

5.1 配置 Kafka 网络连接

5.1.1 连接 Kafka 网络要求

客户端可以通过公网连接Kafka实例，也可以通过内网连接Kafka实例。使用内网连接时，注意以下几点：

- 如果客户端和Kafka实例部署在同一个VPC内，网络默认互通。
- 如果客户端和Kafka实例部署在不同VPC中，由于VPC之间逻辑隔离，客户端和Kafka实例不能直接通信，需要打通VPC之间的网络。

客户端连接Kafka实例的方式如[表5-1](#)所示。

表 5-1 连接方式说明

连接方式	实现方式	参考文档
公网连接	• 客户端通过IPv4地址访问Kafka实例：在Kafka控制台开启公网访问，配置弹性公网IP，客户端通过弹性公网IP访问Kafka实例。 • 客户端通过IPv6地址访问Kafka实例：在Kafka控制台开启IPv6功能，将IPv6地址加入到共享带宽中，实现客户端通过公网访问Kafka实例。	配置Kafka实例的公网访问
	使用DNAT通过端口映射方式，实现客户端通过公网连接Kafka实例。	使用DNAT实现公网访问Kafka
内网连接	客户端和Kafka实例部署在同一个VPC中，此时网络默认互通。	-

连接方式	实现方式	参考文档
	客户端和Kafka实例部署在同一个Region的不同VPC中，利用VPC终端节点在不同VPC间建立跨VPC的连接通道，实现跨VPC访问。	使用VPCPEP实现跨VPC访问Kafka
	客户端和Kafka实例部署在同一个Region的不同VPC中，利用VPC对等连接将两个VPC间的网络打通，实现跨VPC访问。	对等连接

客户端连接Kafka实例前，需要在Kafka实例所属安全组中配置如下安全组规则，否则会连接失败。

 说明

创建安全组后，系统默认添加入方向“允许安全组内的弹性云服务器彼此通信”规则和出方向“放通全部流量”规则，此时使用内网通过同一个VPC访问Kafka实例，无需添加[表5-2](#)的规则。

表 5-2 安全组规则

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9094	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka实例（明文接入）。
入方向	TCP	IPv4	9092	Kafka客户端所在的IP地址或地址组	- 使用内网通过同一个VPC访问Kafka实例（明文接入）。 - 使用对等连接跨VPC访问Kafka实例（明文接入）。
入方向	TCP	IPv6	9192	Kafka客户端所在的IP地址或地址组	通过IPv6地址访问Kafka实例（关闭SSL加密）（内网和公网都适用）。
入方向	TCP	IPv4	9095	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka实例（密文接入）。
入方向	TCP	IPv4	9093	Kafka客户端所在的IP地址或地址组	- 使用内网通过同一个VPC访问Kafka实例（密文接入）。 - 使用对等连接跨VPC访问Kafka实例（密文接入）。
入方向	TCP	IPv6	9193	Kafka客户端所在的IP地址或地址组	通过IPv6地址访问Kafka实例（开启SSL加密）（内网和公网都适用）。

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9011	198.19.128.0/17	通过VPC终端节点实现跨VPC访问Kafka实例（密文接入和明文接入都适用）。
入方向	TCP	IPv4	9011	Kafka客户端所在的IP地址或地址组	使用DNAT访问Kafka实例（密文接入和明文接入都适用）。

5.1.2 配置 Kafka 实例的公网访问

客户端可以通过公网使用IPv4或者IPv6地址访问Kafka实例。

- IPv4地址：在Kafka控制台开启实例的公网访问功能，并设置弹性IP地址，实现公网访问。
- IPv6地址：在创建Kafka实例时开启IPv6功能，然后将IPv6地址添加到共享带宽中，同时支持IPv6内网访问和IPv6公网访问。

约束与限制

Kafka实例只支持绑定IPv4弹性IP地址，不支持绑定IPv6弹性IP地址。

前提条件

- 只有处于“运行中”状态的Kafka实例支持修改公网访问配置。
- （可选）如果使用IPv6地址访问Kafka实例，确认Kafka实例已开启IPv6功能。

开启 Kafka 实例的公网访问

客户端可以通过公网使用IPv4或者IPv6地址访问Kafka实例。

Kafka控制台的“概览 > 连接信息”的内容存在两种不同的显示，导致开启IPv4公网访问的步骤有所不同。

- 已开启IPv6功能，开启IPv4公网访问的步骤请参考“开启IPv4公网访问（不支持修改SASL开关）”。
- 未开启IPv6功能，开启IPv4公网访问的步骤请参考“开启IPv4公网访问（支持修改明文/密文接入）”。

开启 IPv4 公网访问（不支持修改 SASL 开关）

步骤1 登录[Kafka控制台](#)。

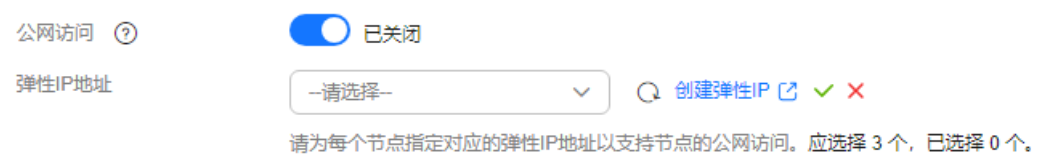
步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例的“概览”页面。

步骤4 在“公网访问”后，单击 ，开启公网访问。在“弹性IP地址”中，为每个代理设置对应的弹性IP地址。

- 如果没有足够数量的弹性IP地址，参考如下步骤设置弹性IP地址。
1. 单击“创建弹性IP”，跳转到“购买弹性公网IP”页面，购买弹性IP地址，具体操作请参考[申请弹性公网IP](#)。
 2. 购买完成后，返回开启公网页面。
 3. 单击“弹性IP地址”后的，然后在下拉框中选择和代理数量相同的弹性IP地址，单击。
 4. 您可以在实例的“实例管理 > 后台任务管理”页面，查看当前任务的操作进度。任务状态为“成功”，表示操作成功。

图 5-1 开启公网访问



开启公网访问后，需要设置对应的[安全组规则](#)，才能成功连接Kafka。连接Kafka的具体操作请参考[连接实例](#)。

表 5-3 Kafka 实例安全组规则（IPv4 公网访问）

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9094	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka（关闭SSL加密）。
入方向	TCP	IPv4	9095	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka（开启SSL加密）。

----结束

开启 IPv4 公网访问（支持修改明文/密文接入）

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4** 在“公网访问”后，单击，开启公网访问。在“弹性IP地址”中，为每个代理设置对应的弹性IP地址。

- 如果没有足够数量的弹性IP地址，参考如下步骤设置弹性IP地址。
1. 单击“创建弹性IP”，跳转到“购买弹性公网IP”页面，购买弹性IP地址，具体操作请参考[申请弹性公网IP](#)。

- 2. 购买完成后，返回开启公网页面。
- 3. 单击“弹性IP地址”后的，然后在下拉框中选择和代理数量相同的弹性IP地址，单击，跳转到“后台任务管理”页面。
- 4. 当任务状态显示为“成功”时，表示成功开启公网访问。

图 5-2 开启公网访问



开启公网访问后，需要设置接入方式（明文接入、密文接入）和对应的安全组规则，才能成功连接Kafka。连接Kafka的具体操作请参考[连接实例](#)。

表 5-4 Kafka 实例安全组规则（IPv4 公网访问）

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9094	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka（明文接入）。
入方向	TCP	IPv4	9095	Kafka客户端所在的IP地址或地址组	通过公网访问Kafka（密文接入）。

----结束

开启 IPv6 公网访问

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 在“连接信息”区域，获取并记录“内网连接地址”的IPv6地址。在“网络”中，获取并记录虚拟私有云和子网。

图 5-3 实例详情页

连接信息

用户名

--

Kafka SASL_SSL

关闭 当前实例暂不支持动态开启/关闭SASL_SSL

内网连接地址

IPv4

192.168.0.169:9092,192.168.0.250:9092,192.168.0.229:9092

IPv6

[2409:2001:0:aa:8fd2:7128:7cc:6105]:9192,[2409:2001:0:aa:a4db:3137:70c9:125a]:9192,[2409:2001:0:aa:5e7d:6b09:5a9b:b9ab]:9192

公网访问

已关闭

VPC内网明文访问

开启

网络

可用区

可用区1

虚拟私有云

vpc-ipv6

子网

subnet-f7ab

安全组

default

IPv6

开启

- 步骤5 在管理控制台左上角单击，选择“网络 > 弹性公网IP EIP”，进入“弹性公网IP”页面。
- 步骤6 在左侧导航栏单击“共享带宽”，进入“共享带宽”页面。
- 步骤7 申请一个共享带宽，具体操作请参考[申请共享带宽](#)。
如果已有共享带宽，可重复使用，无需再次申请。
- 步骤8 在共享带宽所在行，单击“添加公网IP”，弹出“添加公网IP”对话框。
- 步骤9 设置如[表5-5](#)参数，单击“确定”。

表 5-5 添加公网 IP 参数

参数	说明
公网IP	选择“IPv6网卡”
所属VPC	在下拉框中选择 步骤4 中记录的虚拟私有云。
子网	在下拉框中选择 步骤4 中记录的子网，并勾选 步骤4 中记录的全部IPv6地址。

图 5-4 添加公网 IP



步骤10 共享带宽配置成功后，需要在Kafka实例的安全组中设置如表5-6所示规则，才能成功通过IPv6地址连接Kafka。

表 5-6 Kafka 实例安全组规则（IPv6 访问）

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv6	9192	::/0	通过IPv6地址访问Kafka实例（关闭SSL加密）。
入方向	TCP	IPv6	9193	::/0	通过IPv6地址访问Kafka实例（开启SSL加密）。

客户端通过IPv6的公网连接Kafka实例时，注意以下几点：

- Kafka的连接地址为“内网连接地址”中的IPv6地址。
- 客户端网卡必须添加到共享带宽中。客户端网卡所在的共享带宽和Kafka实例所在的共享带宽不需要为同一个，共享带宽间网络默认互通。

----结束

关闭 Kafka 实例的公网访问

Kafka控制台的“概览 > 连接信息”的内容存在两种不同的显示，导致关闭IPv4公网访问的步骤有所不同。

- 已开启IPv6功能，关闭Pv4公网访问的步骤请参考“关闭IPv4公网访问（不支持修改SASL开关）”。
- 未开启IPv6功能，关闭IPv4公网访问的步骤请参考“关闭IPv4公网访问（支持修改明文/密文接入）”。

关闭 IPv4 公网访问（不支持修改 SASL 开关）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 在“公网访问”后，单击，完成公网访问的关闭。

您可以在实例的“实例管理 > 后台任务管理”页面，查看当前任务的操作进度。任务状态为“成功”，表示操作成功。

关闭公网访问后，需要设置对应的[安全组规则](#)，才能通过内网成功连接Kafka。连接Kafka的具体操作请参考[连接实例](#)。

表 5-7 Kafka 实例安全组规则（内网访问）

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9092	Kafka客户端所在的IP地址或地址组	使用内网通过同一个VPC访问Kafka实例（关闭SSL加密）。
入方向	TCP	IPv4	9093	Kafka客户端所在的IP地址或地址组	使用内网通过同一个VPC访问Kafka实例（开启SSL加密）。

说明

创建安全组后，系统默认添加入方向“允许安全组内的弹性云服务器彼此通信”规则和出方向“放通全部流量”规则，此时使用内网通过同一个VPC访问Kafka实例，无需添加[表5-7](#)的规则。

----结束

关闭 IPv4 公网访问（支持修改明文/密文接入）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 关闭公网访问前，需要先关闭公网访问中的“明文接入”和“密文接入”。然后在“公网访问”后，单击，弹出确认关闭对话框。
- 步骤5 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功关闭公网访问。

关闭公网访问后，需要设置对应的[安全组规则](#)，才能成功通过内网连接Kafka。连接Kafka的具体操作请参考[连接实例](#)。

 说明

创建安全组后，系统默认添加入方向“允许安全组内的弹性云服务器彼此通信”规则和出方向“放通全部流量”规则，此时使用内网通过同一个VPC访问Kafka实例，无需添加[表5-8](#)的规则。

表 5-8 Kafka 实例安全组规则（内网访问）

方向	协议	类型	端口	源地址	说明
入方向	TCP	IPv4	9092	Kafka客户端所在的IP地址或地址组	使用内网通过同一个VPC访问Kafka实例（明文接入）。
入方向	TCP	IPv4	9093	Kafka客户端所在的IP地址或地址组	使用内网通过同一个VPC访问Kafka实例（密文接入）。

----结束

关闭 IPv6 公网访问

将Kafka实例的IPv6地址从共享带宽中移出，具体请参考[从共享带宽中移出弹性公网IP](#)。

5.1.3 使用 VPCEP 实现跨 VPC 访问 Kafka

Kafka客户端和Kafka实例在同一区域的不同VPC中，由于VPC之间逻辑隔离，Kafka客户端和Kafka实例不能直接通信。您可以通过以下任意一个方式实现跨VPC访问：

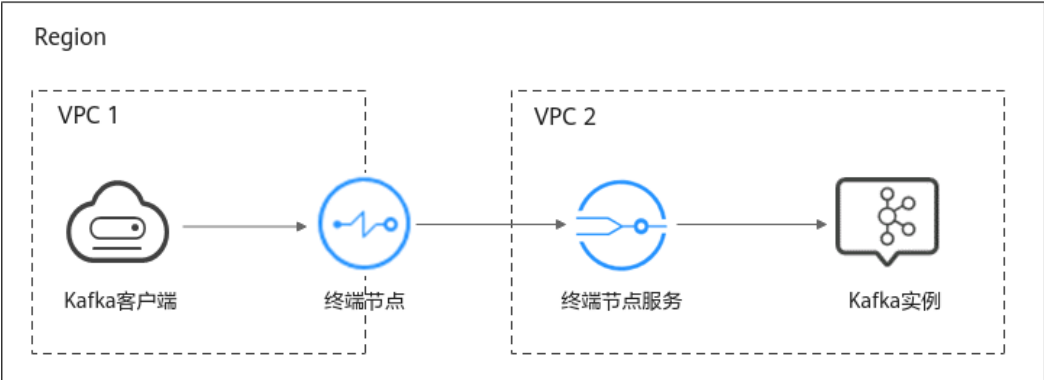
- 创建VPC对等连接，将两个VPC的网络打通，实现跨VPC访问。具体步骤请参考[对等连接](#)。
- 利用VPC终端节点在不同VPC间建立跨VPC的连接通道，实现Kafka客户端通过内网访问Kafka实例。

本章节主要介绍通过VPC终端节点实现跨VPC访问的方法。

VPC终端节点由“终端节点服务”和“终端节点”两种资源实例组成。

- 终端节点服务：将Kafka实例配置为VPC终端节点支持的服务，可以被终端节点连接和访问。
- 终端节点：用于在VPC和终端节点服务之间建立便捷、安全、私密的连接通道。

图 5-5 跨 VPC 访问 Kafka 实例



客户端使用 VPCEP 跨 VPC 访问 Kafka 时，使用的是明文连接还是密文连接？

使用明文连接还是密文连接，取决于“跨VPC访问协议”。跨VPC访问协议是在创建Kafka实例时设置的，实例创建成功后，不支持修改。

跨VPC访问协议的取值如下：

- PLAINTEXT：表示客户端访问Kafka实例时，无需认证，数据通过明文传输。
- SASL_SSL：表示客户端访问Kafka实例时，使用SASL认证，数据通过SSL证书加密传输。
- SASL_PLAINTEXT：表示客户端访问Kafka实例时，使用SASL认证，数据通过明文传输。

步骤一：创建终端节点服务

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在“概览”页面的“高级配置”区域，获取Kafka实例的网卡地址（listeners IP）和对应的Port ID。

图 5-6 Kafka 实例的网卡地址（listeners IP）和对应的 Port ID

跨VPC访问 ⓘ 修改

listeners IP	advertised.listeners IP/域名	Port	Port ID
192.168.0.25	192.168.0.25	9011	cbd7904e-2728-4458-b44d-0a105 
192.168.0.174	192.168.0.174	9011	29c29170-4273-4230-b44d-0a1051d18 
192.168.0.70	192.168.0.70	9011	52fa7904-2728-4458-b44d-0a10503 

- 步骤5 在“概览”页面的“网络”区域，查看Kafka实例所在的VPC名称。

图 5-7 查看 Kafka 实例所在的 VPC 名称



步骤6 单击VPC名称，进入VPC基本信息页签，获取VPC ID。

图 5-8 获取 VPC ID



步骤7 调用VPC终端节点的API创建终端节点服务。参考表5-9设置请求参数，其他参数根据实际情况设置。API详情请参考创建终端节点服务。

```
POST https://{endpoint}/v1/{project_id}/vpc-endpoint-services
```

表 5-9 创建终端节点服务参数说明

参数名称	说明
port_id	终端节点服务后端资源的ID。 输入步骤4中获取的其中一个Port ID。
vpc_id	终端节点服务对应后端资源所在的VPC的ID。 输入步骤6中获取的VPC ID。
server_type	资源类型。 输入“VM”。
client_port	终端节点访问的端口。 输入“9011”。
server_port	终端节点服务的端口。 输入“9011”。

参数名称	说明
protocol	端口映射协议。 输入“TCP”。
approval_enabled	是否需要审批。 输入“false”，表示不需要审批，创建的终端节点连接直接为accepted状态。
service_type	终端节点服务类型。 输入“interface”。
endpoint	获取VPC终端节点的终端节点，区域必须与Kafka实例保持一致。
project_id	获取VPC终端节点的项目ID，区域必须与Kafka实例保持一致。

记录响应信息中的“service_name”参数值，此参数表示终端节点服务的名称。

步骤8 参考[步骤7](#)，为[步骤4](#)中其他Port ID创建终端节点服务，并记录终端节点服务的名称。
----结束

（可选）步骤二：添加白名单

VPCEP允许跨账号的终端节点连接终端节点服务，通过设置终端节点服务的白名单实现。

Kafka客户端和Kafka实例属于不同账号时，将Kafka客户端所在账号的ID添加到终端节点服务的白名单中，完成跨账号终端节点的访问授权，具体操作步骤请参考[添加白名单](#)。

步骤三：购买终端节点

- 步骤1** 在管理控制台左上角单击，选择“网络 > VPC终端节点”，进入终端节点页面。
- 步骤2** 单击“购买终端节点”，进入“购买终端节点”页面。
- 步骤3** 参考[表5-10](#)设置参数，其他参数保持默认，如果想要了解更多的参数信息，请参考[购买终端节点](#)。

图 5-9 终端节点参数设置

* 区域

不同区域的云服务产品之间网互不相通，请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。

* 计费模式

按需计费

* 服务类别

云服务

按名称查找服务

* 服务名称

521560-efd2-4bb8-84ed-5d707302

验证

已找到服务

服务类型：接口

☒ 创建内网域名

* 虚拟私有云

vpc-kafka(192.168.0.0/16)

查看虚拟私有云

* 子网

subnet-kafka(192.168.0.0/...

查看已有子网

可用IP数：248

* IPv4地址

自动分配IPv4地址

手动指定IP地址

访问控制

标签

如果您需要使用同一标签标识多种云资源，即所有服务均可在标签输入框下拉选择同一标签，建议在TMS中创建预定义标签。[查看预定义标签](#)

标签键

标签值

您还可以添加20个标签。

描述

0/512

表 5-10 创建终端节点参数说明

参数名称	说明
区域	终端节点所在区域。与Kafka实例保持一致。
服务类别	- 云服务：当您要连接的终端节点服务为云服务时，需要选择“云服务”。 - 按名称查找服务：当您要连接的终端节点服务为用户私有服务时，需要选择“按名称查找服务”。 选择“按名称查找服务”。
服务名称	输入步骤7中记录的终端节点服务名称，单击“验证”。显示“已找到服务”后，继续后续操作。
（可选）创建内网域名	如果您想以域名方式访问终端节点，请勾选“创建内网域名”。终端节点创建完成后，便能通过内网域名直接访问终端节点。
虚拟私有云	终端节点所属的虚拟私有云。选择Kafka客户端所属的VPC。
子网	终端节点所属的子网。选择Kafka客户端所属的子网。
IPv4地址	终端节点的IPv4地址。选择“自动分配IPv4地址”。

步骤4 单击“立即购买”，进入规格确认页面。

- 步骤5** 确认无误后，提交请求。
- 步骤6** 购买成功后，返回终端节点页面，查看终端节点状态是否为“已接受”，“已接受”表示终端节点已成功连接至终端节点服务。

图 5-10 查看终端节点状态

ID	监控	虚拟私...	状态	终端节点服务名称	类型	服务地址	创建时间
5f285b1e-bee4-4bc8...		vpc-kafka	已接受	521560...	接口	192.168.0.74	2024/05/20 ...

- 步骤7** 单击终端节点ID，在“基本信息”页签，查看并记录节点IP。您可以使用该IP访问终端节点，进行跨VPC资源通信。
- 如果您想通过域名访问终端节点，进行跨VPC资源通信，在“基本信息”页签查看并记录内网域名。

 **注意**

复制内网域名时，不要复制最后一个点。正确的域名格式为“vpcep-5xxx7-e302-4ab4-bc6c-54xxxe52.xx-xx-xx.xxx.com”。

图 5-11 查看节点 IP

基本信息 访问控制 监控 标签			
ID	5f285b1e-bee4-4bc8...	状态	已接受
虚拟私有云	vpc-kafka	类型	接口
付费方	服务使用方	终端节点服务名称	vpcep-5f285b1e-bee4-4bc8...
IPv4地址	192.168.0.74	创建时间	2024/05/20 09:44:30 GMT+08:00
访问控制	☐	内网域名	vpcep-5f285b1e-bee4-4bc8...-e302-4ab4-bc6c-54xxxe52.xx-xx-xx.xxx.com
描述	-		

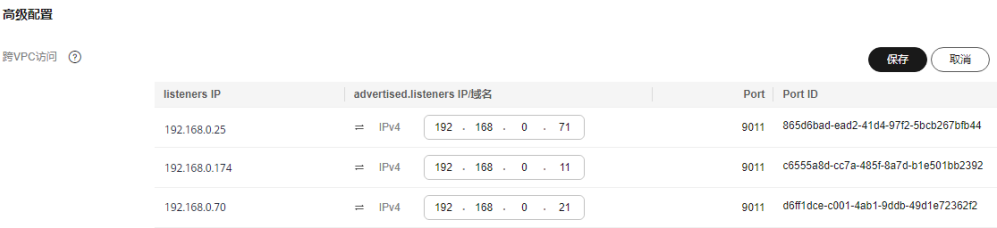
- 步骤8** 参考**步骤1~步骤7**，为**步骤8**中创建的终端节点服务购买终端节点，查看并记录节点IP。
- 如果您想通过域名访问终端节点，请记录内网域名。
- 结束

步骤四：修改 advertised.listeners IP

- 步骤1** 在管理控制台左上角单击，选择“应用中间件 > 分布式消息服务Kafka版”，进入Kafka总览页面。
- 步骤2** 在左侧导航栏单击“Kafka实例”，进入Kafka实例列表页面。
- 步骤3** 单击Kafka实例名称，进入实例详情页面。
- 步骤4** 在“概览”页面的“高级配置”区域，单击“修改”，将“跨VPC访问”的“advertised.listeners IP/域名”修改为**步骤7**和**步骤8**中记录的节点IP。节点IP必须与Port ID一一对应，否则会导致网络不通。修改完后，单击“保存”。

如果您想通过域名访问终端节点，在“概览”页面的“高级配置”区域，单击“修改”。在“advertised.listeners IP/域名”列，单击，切换到域名模式。将“advertised.listeners IP/域名”修改为**步骤7**和**步骤8**中记录的内网域名。内网域名必须与Port ID一一对应，否则会导致网络不通。

图 5-12 修改 advertised.listeners IP



----结束

步骤五：验证接口连通性

参考[使用客户端连接Kafka（明文接入）](#)或者[使用客户端连接Kafka（密文接入）](#)，测试是否可以生产和消费消息。

测试接口连通性时，注意以下几点：

- 连接Kafka实例的地址为“advertised.listeners IP:9011”，以图5-12为例，连接Kafka实例的地址为“192.168.0.71:9011,192.168.0.11:9011,192.168.0.21:9011”。
如果使用域名访问终端节点，连接Kafka实例的地址为“域名:9011”，例如“vpcep-5xxx7-e302-4ab4-bc6c-54xxxe52.xx-xx-xx.xxx.com:9011”。
- 在Kafka实例安全组的入方向规则中放通9011端口，以及198.19.128.0/17网段的地址。
- 如果Kafka实例的子网配置了网络ACL功能，需要在网络ACL的入方向规则中放通198.19.128.0/17网段的地址，以及VPC终端节点涉及的子网。

说明

198.19.128.0/17是为VPC终端节点分配的网段，使用VPC终端节点需要放通此网段。

5.1.4 使用 DNAT 实现公网访问 Kafka

您可以通过以下任意一个方式实现公网访问：

- 在Kafka实例控制台开启公网访问，使用弹性IP地址连接Kafka实例，具体请参考[配置Kafka实例的公网访问](#)。
- 使用DNAT通过端口映射方式，将弹性IP地址映射到Kafka实例指定端口，实现Kafka实例对公网提供服务。

本章节主要介绍使用DNAT实现公网访问Kafka。

前提条件

已购买弹性公网IP，弹性公网IP的数量与Kafka实例中代理个数相同。购买弹性IP地址的操作指导请参考[申请弹性公网IP](#)。

步骤一：获取 Kafka 实例的信息

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在“概览”页面的“连接信息”区域，获取并记录Kafka实例的内网连接地址。在“网络”区域，获取并记录Kafka实例所在的虚拟私有云和子网。

图 5-13 Kafka 实例信息



----结束

步骤二：购买公网 NAT 网关

- 步骤1 在管理控制台左上角单击，选择“网络 > NAT网关”，进入“公网NAT网关”页面。
- 步骤2 单击“购买公网NAT网关”，进入“购买公网NAT网关”页面。
- 步骤3 参考表5-11设置参数，其他参数请根据实际情况填写，如果想要了解更多的参数信息，请参考购买公网NAT网关。

图 5-14 购买公网 NAT 网关



表 5-11 创建公网 NAT 网关参数说明

参数名称	说明
区域	公网NAT网关所在的区域。与Kafka实例保持一致。
名称	您自定义的公网NAT网关名称。最大支持64个字符，仅支持中文、数字、字母、_（下划线）、-（中划线）、.（点号）。
虚拟私有云	公网NAT网关所属的VPC。选择 获取Kafka实例的信息 中记录的虚拟私有云。
子网	公网NAT网关所属VPC中的子网。选择 获取Kafka实例的信息 中记录子网。
企业项目	公网NAT网关归属的企业项目。根据实际情况选择。

- 步骤4** 单击“立即购买”，进入规格确认页面。
- 步骤5** 确认规格无误后，如果“计费模式”选择“包年/包月”，单击“去支付”。根据界面提示信息，支付费用。如果“计费模式”选择“按需付费”，单击“提交”。
- 结束

步骤三：添加 DNAT 规则

- 步骤1** 在“公网NAT网关”页面，在新购买的公网NAT网关后，单击“设置规则”，进入公网NAT网关详情页。
- 步骤2** 在“DNAT规则”页签，单击“添加DNAT规则”，弹出“添加DNAT规则”对话框。

图 5-15 公网 NAT 网关详情页



- 步骤3** 参考[表5-12](#)设置参数，如果想要了解更多的参数信息，请参考[添加DNAT规则](#)。

图 5-16 添加 DNAT 规则

添加DNAT规则

NAT网关名称

nat-kafka

★ 使用场景

虚拟私有云

云专线/云连接

★ 端口类型

具体端口

所有端口

★ 支持协议

TCP

▼

★ 公网IP类型

弹性公网IP

全域弹性公网IP

120 54(1 Mbit/s | 按需计费 | default)

▼

查看弹性公网IP

?

带宽大小: 1 Mbit/s 计费模式: 按需计费

企业项目: default

★ 公网端口

9011

★ 实例类型

服务器

虚拟IP地址

自定义

★ 私网IP

-

-

-

★ 私网端口

9011

描述

表 5-12 添加 DNAT 规则

参数名称	说明
使用场景	选择“虚拟私有云”，表示虚拟私有云中的云主机将通过DNAT的方式共享弹性公网IP，为公网提供服务。
端口类型	选择“具体端口”，公网NAT网关会将以指定协议和端口访问该弹性公网IP的请求转发到目标云主机实例的指定端口上。
支持协议	选择“TCP”。
公网IP类型	用来提供互联网访问的公网IP的类型。 选择“弹性公网IP”，在下拉框中选择已购买的弹性公网IP。
公网端口	弹性公网IP的端口，输入“9011”。
实例类型	对外部公网提供服务的实例类型，选择“自定义”。
私网IP	输入 获取Kafka实例的信息 中记录的Kafka实例的一个内网连接地址。
私网端口	私网IP的端口，输入“9011”。

步骤4 单击“确定”，完成DNAT规则的添加。

DNAT规则添加成功后，在DNAT规则列表中查看此规则的状态，若“状态”为“运行中”，表示创建成功。

步骤5 为[获取Kafka实例的信息](#)中记录的其他内网连接地址创建DNAT规则，每个DNAT规则需要设置不同的弹性公网IP。

创建DNAT规则的具体步骤参考[步骤2~步骤4](#)。

步骤6 DNAT规则全部创建成功后，在“DNAT规则”页签，查看已创建的DNAT规则，并记录私网IP对应的弹性公网IP。

图 5-17 DNAT 规则列表

ID	状态	使用场景	关联的公网IP	公网端口	私网IP	私网端口	支持协议
2bed5dc1-2954-466...	运行中	虚拟私有云	100.100.0.7	9011	192.168.0.96	9011	TCP
198139a8-b66b-48...	运行中	虚拟私有云	100.100.0.123	9011	192.168.0.64	9011	TCP
8d9feafb-8aa9-47d...	运行中	虚拟私有云	100.100.0.254	9011	192.168.0.235	9011	TCP

----结束

步骤四：将弹性公网 IP 地址映射到内网 IP 地址的 9011 端口

步骤1 在管理控制台左上角单击，选择“应用中间件 > 分布式消息服务Kafka版”，进入Kafka总览页面。

步骤2 在左侧导航栏单击“Kafka实例”，进入Kafka实例列表页面。

步骤3 单击Kafka实例名称，进入实例详情页面。

步骤4 在“概览”页面的“高级配置”区域，单击“修改”。

步骤5 将“advertised.listeners IP/域名”改为DNAT规则中的弹性公网IP，内网连接地址和弹性公网IP的对应关系与[添加DNAT规则](#)中记录的对应关系保持一致，单击“保存”。

图 5-18 修改 advertised.listeners IP（使用 DNAT 访问）

高级配置

跨VPC访问 ⓘ

保存 取消

listeners IP	advertised.listeners IP/域名	Port	Port ID
192.168.0.96	= IPv4 1 . . . 197	9011	1eee4c5b-63c9-47c5-b618-696bbe01c9a9
192.168.0.64	= IPv4 123 . . . 123	9011	a0011065-88dc-44bd-a6d9-97d19a46d36c
192.168.0.235	= IPv4 120 . . . 254	9011	825fc783-f937-4c5e-9c2a-76da0a651cb3

----结束

步骤五：验证接口连通性

参考[使用客户端连接Kafka（明文接入）](#)或者[使用客户端连接Kafka（密文接入）](#)，测试是否可以生产和消费消息。

测试接口连通性时，注意以下几点：

- 连接Kafka实例的地址为“advertised.listeners IP:9011”，以[图5-18](#)为例，连接Kafka实例的地址为“124.xxx.xxx.167:9011,124.xxx.xxx.174:9011,124.xxx.xxx.57:9011”。

- 在Kafka实例安全组的入方向规则中放通**9011**端口。
- 连接Kafka实例的客户端已开启公网访问功能。

5.2 配置 Kafka 访问控制

5.2.1 配置 Kafka 实例的明文/密文接入

您可以通过明文接入Kafka实例，也可以通过密文接入Kafka实例，本章节指导您在控制台修改接入方式。

- 明文接入：表示客户端连接Kafka实例时，无需进行SASL认证。
- 密文接入：表示客户端连接Kafka实例时，需要进行SASL认证。

约束与限制

- 首次修改接入方式时，部分实例会重启，另一部分实例不会重启，具体以控制台提示为准。实例重启大约需要75~80s。后续再次修改接入方式时，实例不会再重启。
- 单机实例只支持开启/关闭公网访问的明文接入。
- 开启了IPv6功能后，不支持修改实例的接入方式。

前提条件

只有处于“运行中”状态的Kafka实例才可以修改接入方式。

配置 Kafka 实例的明文接入

明文接入表示客户端连接Kafka实例时，无需进行SASL认证。

开启明文接入

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4** 明文接入分为内网访问明文接入和公网访问明文接入，开启明文接入方法如[表5-13](#)所示。

表 5-13 开启明文接入方法

接入方式	开启明文接入
内网访问明文接入	1. 在“内网访问 > 明文接入”后，单击 ，弹出确认对话框。 2. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功开启明文接入。

接入方式	开启明文接入
公网访问明文接入	1. 确认公网访问已开启，如果未开启，请开启公网访问，具体步骤参考配置Kafka实例的公网访问。 2. 在“公网访问 > 明文接入”后，单击，弹出确认对话框。 3. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功开启明文接入。

----结束

关闭明文接入

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 明文接入分为内网访问明文接入和公网访问明文接入，关闭明文接入方法如[表5-14](#)所示。

表 5-14 关闭明文接入方法

接入方式	关闭明文接入
内网访问明文接入	内网访问不支持关闭，明文接入和密文接入至少开启一个。如果密文接入没有开启，此时无法关闭明文接入。 1. 在“内网访问 > 明文接入”后，单击，弹出确认对话框。 2. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功关闭明文接入。
公网访问明文接入	1. 在“公网访问 > 明文接入”后，单击，弹出确认对话框。 2. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功关闭明文接入。

----结束

配置 Kafka 实例的密文接入

密文接入表示客户端连接Kafka实例时，需要进行SASL认证。

开启密文接入

- 步骤1 登录[Kafka控制台](#)。

- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 密文接入分为内网访问密文接入和公网访问密文接入，开启密文接入方法如表5-15所示。

表 5-15 开启密文接入方法

接入方式	开启密文接入
内网访问密文接入	1. 在“内网访问 > 密文接入”后，单击 ，弹出“内网访问密文接入”对话框。 2. 设置kafka安全协议、SASL PLAIN机制、用户名和密码，单击“确定”。跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功开启密文接入。 说明 首次开启密文接入（包括内网访问密文接入和公网访问密文接入）时，需要设置kafka安全协议、SASL PLAIN机制、用户名和密码，再次开启时，只需要设置kafka安全协议。
公网访问密文接入	1. 确认公网访问已开启，如果未开启，请开启公网访问，具体步骤参考配置Kafka实例的公网访问。 2. 在“公网访问 > 密文接入”后，单击 ，弹出“公网访问密文接入”对话框。 3. 设置kafka安全协议、SASL PLAIN机制、用户名和密码，单击“确定”。跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功开启密文接入。 说明 首次开启密文接入（包括内网访问密文接入和公网访问密文接入）时，需要设置kafka安全协议、SASL PLAIN机制、用户名和密码，再次开启时，只需要设置kafka安全协议。

kafka安全协议、SASL PLAIN机制、用户名和密码的参数解释如下：

表 5-16 密文接入参数说明

参数名称	参数值	说明
kafka安全协议	SASL_SSL	采用SASL方式进行认证，数据通过SSL证书进行加密传输，安全性更高。
	SASL_PLAINTEXT	采用SASL方式进行认证，数据通过明文传输，性能更好。 由于明文传输，建议使用SCRAM-SHA-512机制。

参数名称	参数值	说明
SASL PLAIN机制	-	- 未开启“SASL PLAIN机制”时，使用SCRAM-SHA-512机制对账号密码进行认证。 - 开启“SASL PLAIN机制”后，同时支持SCRAM-SHA-512机制和PLAIN机制，根据实际情况选择其中任何一种配置连接。 密文接入成功开启后，SASL PLAIN机制不支持修改。 什么是SCRAM-SHA-512机制和PLAIN机制？ - SCRAM-SHA-512机制：采用哈希算法对用户名与密码生成凭证，进行身份校验的安全认证机制，比PLAIN机制安全性更高。 - PLAIN机制：一种简单的用户名密码校验机制。
用户名、密码	-	客户端用于连接Kafka实例的用户名和密码。 用户名需要符合以下命名规则：由英文字母开头，且只能由英文字母、数字、中划线、下划线组成，长度为4~64个字符。 密码需要符合以下规则： - 长度为8~32个字符。 - 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=+ [{}];:","<.>?`和空格，并且不能以-开头。 - 不能与用户名或倒序的用户名相同。 密文接入成功开启后，用户名不支持修改。

kafka安全协议、SASL PLAIN机制、用户名和密码在客户端连接已开启密文接入的Kafka实例时需要配置，具体请参考[使用客户端连接Kafka（密文接入）](#)。

----结束

关闭密文接入

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 密文接入分为内网访问密文接入和公网访问密文接入，关闭密文接入方法如[表5-17](#)所示。

表 5-17 关闭密文接入方法

接入方式	关闭明文接入
内网访问密文接入	1. 在“内网访问 > 密文接入”后，单击，弹出确认对话框。 2. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功关闭密文接入。
公网访问密文接入	1. 在“公网访问 > 密文接入”后，单击，弹出确认对话框。 2. 单击“确定”，跳转到“后台任务管理”页面。当任务状态显示为“成功”时，表示成功关闭密文接入。

关闭内网访问密文接入和公网访问密文接入后，已创建的用户不会被删除，下次开启密文接入时，无需再次创建用户。

----结束

5.2.2 制作和替换 Kafka JKS 格式的 SSL 证书

SSL证书的作用是在客户端和Kafka实例进行数据传输时对数据加密传输，提高了数据的安全性。

Kafka客户端连接开启密文接入且Kafka安全协议为“SASL_SSL”的Kafka实例时，您可以使用分布式消息服务Kafka版提供的证书，也可以使用自己制作的证书。如果使用自己制作的证书，请参考本章节将分布式消息服务Kafka版提供的证书替换为自己的证书。

制作并替换证书为白名单特性，需要联系后台管理人员开通后方可使用。所有region都可以通过白名单开通此特性。

 注意

[步骤一~步骤四](#)主要介绍如何制作适用于未开启证书域名校验场景的SSL证书，并完成证书替换操作。如果您需要制作适用于开启证书域名校验场景的SSL证书，请参考[（可选）制作和替换开启证书域名校验的SSL证书](#)。

约束与限制

替换证书将重启实例，请谨慎操作。

前提条件

- 已准备一台Linux系统的服务器，并安装[Java Development Kit 1.8.111或以上版本](#)，配置JAVA_HOME与PATH环境变量。
- 实例已开启Kafka SASL_SSL。
- （可选）如果制作开启证书域名校验的SSL证书，请在Kafka实例详情页的“连接信息”中获取连接地址。

步骤一：制作证书

步骤1 登录Linux系统的服务器，执行以下命令，生成server.keystore.jks证书。

```
keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA
```

按照提示信息输入keystore密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}';",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

按照提示信息输入证书拥有者的信息，包含姓名、单位、组织名称、所在城市、所在省、所在国家，示例如下。

```
[root@ecs-kafka ~]# keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Tom
What is the name of your organizational unit?
[Unknown]: test
What is the name of your organization?
[Unknown]: test01
What is the name of your City or Locality?
[Unknown]: nj
What is the name of your State or Province?
[Unknown]: js
What is the two-letter country code for this unit?
[Unknown]: xx
Is CN=Tom, OU=test, O=test01, L=nj, ST=js, C=xx correct?
[no]: y
```

步骤2 执行以下命令，创建CA的认证证书。

```
openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
```

按照提示信息输入PEM密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：4~1024个字符。

按照提示信息输入证书拥有者的信息，包含所在国家、所在省、所在城市、组织名称、单位、姓名、邮箱，示例如下。

```
[root@ecs-kafka ~]# openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'ca-key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:xx
State or Province Name (full name) []:js
Locality Name (eg, city) [Default City]:nj
Organization Name (eg, company) [Default Company Ltd]:test01
Organizational Unit Name (eg, section) []:test
```

```
Common Name (eg, your name or your server's hostname) []:Tom
Email Address []:xx
[root@ecs-kafka ~]#
```

步骤3 执行以下命令，通过CA创建一个服务器的Truststore证书，有了Truststore证书才可以进行证书有效性的检测。

```
keytool -keystore server.truststore.jks -alias CARoot -import -file ca-cert
```

按照提示信息输入服务器证书的Truststore密码，并记录此密码。此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤4 执行以下命令，通过CA创建一个客户端的Truststore证书。

```
keytool -keystore client.truststore.jks -alias CARoot -import -file ca-cert
```

按照提示信息输入客户端证书的Truststore密码，并记录此密码。此密码用于客户端连接Kafka实例时，配置文件中“ssl.truststore.password”参数的值。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤5 执行以下命令，为服务器证书签名。

1. 导出服务器证书“server.cert-file”。

```
keytool -keystore server.keystore.jks -alias localhost -certreq -file server.cert-file
```

按照提示信息输入[步骤1](#)中设置的keystore密码。

2. 通过CA给服务器证书进行签名处理。

```
openssl x509 -req -CA ca-cert -CAkey ca-key -in server.cert-file -out server.cert-signed -days 3650 -CAcreateserial
```

按照提示信息输入[步骤2](#)中设置的PEM密码。

3. 导入CA证书到服务器keystore。

```
keytool -keystore server.keystore.jks -alias CARoot -import -file ca-cert
```

按照提示信息输入[步骤1](#)中设置的keystore密码。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

4. 将已签名的服务器证书导入到服务器keystore。

```
keytool -keystore server.keystore.jks -alias localhost -import -file server.cert-signed
```

按照提示信息输入[步骤1](#)中设置的keystore密码。

步骤6 导出证书 “server.keystore.jks”、“server.truststore.jks” 和 “client.truststore.jks” 到本地PC机。

“server.keystore.jks” 和 “server.truststore.jks” 在后续步骤 “替换证书” 时，用来替换 “Keystore文件” 和 “Truststore文件”。请将 “client.truststore.jks” 存放到客户端指定位置中，并记录其存放路径，此路径将作为客户端连接Kafka实例时，配置文件中 “ssl.truststore.location” 参数的值。

图 5-19 证书目录

```
total 44
drwxr-xr-x  2 root root 4096 Aug 10 15:20 ./
drwxr-xr-x 10 root root 4096 Aug  8 17:04 ../
-rw-r--r--  1 root root 1322 Aug  8 17:07 ca-cert
-rw-r--r--  1 root root   41 Aug  8 17:09 ca-cert.srl
-rw-----  1 root root 1854 Aug  8 17:07 ca-key
-rw-r--r--  1 root root 1226 Aug  8 17:08 client.truststore.jks
-rw-r--r--  1 root root 1055 Aug  8 17:09 server.cert-file
-rw-r--r--  1 root root 1176 Aug  8 17:09 server.cert-signed
-rw-r--r--  1 root root 4693 Aug  8 17:10 server.keystore.jks
-rw-r--r--  1 root root 1226 Aug  8 17:08 server.truststore.jks
```

----结束

步骤二：替换证书

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击待替换证书的实例名称，进入实例详情页面。

步骤4 在“连接信息 > SSL证书”后，单击“替换”，弹出“替换SSL证书”对话框。

图 5-20 连接信息

连接信息

用户名

root [重置密码](#)

内网访问

明文接入

☒

内网明文连接地址

192.168.79.91:9092,192.168.100.... [📄](#)

密文接入

☐

公网访问 [?](#)

☒ 已开启

明文接入

☒

公网明文连接地址

100... 204:9094,100... [📄](#)

密文接入

☒

公网密文连接地址

100... 204:9095,100... [📄](#)

启用的安全协议

SASL_SSL

开启的SASL认证机制

SCRAM-SHA-512

SSL 双向认证

☐ 关闭

SSL 证书

[下载](#) | [替换](#)

步骤5 参考表5-18，设置替换SSL证书的参数。

图 5-21 替换 SSL 证书

替换 SSL 证书

替换证书将重启实例。

×

Key 密码

Keystore 密码

Keystore 文件

添加文件

只支持 jks 文件且大小不超过 100 KB

Truststore 密码

Truststore 文件

添加文件

只支持 jks 文件且大小不超过 100 KB

表 5-18 替换 SSL 证书参数说明

参数名称	说明
Key密码	输入制作证书中设置的keystore密码。
Keystore密码	输入制作证书中设置的keystore密码。
Keystore文件	导入“server.keystore.jks”证书。
Truststore密码	输入服务端证书的Truststore密码。
Truststore文件	导入“server.truststore.jks”证书。

- 步骤6 单击“确定”，弹出“替换SSL证书”对话框。
- 步骤7 单击“确认”，完成证书的替换。
- 在“实例管理 > 后台任务管理”页面，替换SSL证书任务的“状态”为“成功”时，表示替换证书成功。
- 证书替换成功后，在实例详情页单击“下载”，下载的证书为分布式消息服务Kafka版提供的证书，并非您自己制作的证书。
- 结束

步骤三：修改客户端配置文件

替换证书后，需要在客户端的“consumer.properties”和“producer.properties”文件中，分别修改“ssl.truststore.location”和“ssl.truststore.password”参数。

```
security.protocol=SASL_SSL
ssl.truststore.location=/opt/kafka_2.12-2.7.2/config/client.truststore.jks
ssl.truststore.password=xxxxb
ssl.endpoint.identification.algorithm=
```

表 5-19 配置文件参数说明

参数名称	说明
ssl.truststore.location	client.truststore.jks 证书的存放路径。
ssl.truststore.password	客户端证书的Truststore密码 。
ssl.endpoint.identification.algorithm	证书域名校验开关，为空则表示关闭。这里需要保持关闭状态，必须设置为空。

步骤四：验证证书是否生效

参考[使用客户端连接Kafka（密文接入）](#)，生产和消费消息。如果成功，表示替换的证书已生效。

（可选）制作和替换开启证书域名校验的 SSL 证书

警告

开启证书域名校验后，变更连接实例的地址，例如：内网连接地址变为公网连接地址、公网连接地址变为内网连接地址、扩容代理数量后连接地址增加等，可能会导致客户端连接实例失败。

步骤1 登录Linux系统的服务器，执行以下命令，生成server.keystore.jks证书。

```
keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA -ext
SAN=IP:xxx.xxx.xx.xx,IP:xxx.xxx.xx.xx,IP:xxx.xxx.xx.xx
```

上述命令中的IP地址为[Kafka实例连接地址](#)中的IP地址，从[前提条件](#)中获取。注意：Kafka实例的连接地址包含IP地址:端口号，例如：192.168.10.10:9093,192.168.10.11:9093,192.168.10.12:9093，上述命令需要修改为：

```
keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA -ext
SAN=IP:192.168.10.10,IP:192.168.10.11,IP:192.168.10.12
```

1. 执行上述命名后，按照提示信息输入keystore密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#%&\*()-\_+=\|[]{}';",<.>/?`和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

- 按照提示信息输入证书拥有者的信息，包含姓名、单位、组织名称、所在城市、所在省、所在国家，示例如下。

```
[root@ecs-kafka ~]# keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -
keyalg RSA -ext SAN=IP:192.168.10.10,IP:192.168.10.11,IP:192.168.10.12
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Tom
What is the name of your organizational unit?
[Unknown]: test
What is the name of your organization?
[Unknown]: test01
What is the name of your City or Locality?
[Unknown]: nj
What is the name of your State or Province?
[Unknown]: js
What is the two-letter country code for this unit?
[Unknown]: xx
Is CN=Tom, OU=test, O=test01, L=nj, ST=js, C=xx correct?
[no]: y
```

- 按照提示信息按“Enter”，设置key密码与keystore密码一致。

```
Enter key password for <localhost>
(RETURN if same as keystore password):
```

步骤2 执行以下命令，生成CA私钥。

```
openssl genrsa -out ca.key 2048
```

步骤3 执行以下命令，生成CA证书。

```
openssl req -new -x509 -key ca.key -out ca.crt -days 3650 -subj "/CN=KafkaTestCA"
```

步骤4 执行以下命令，生成服务端证书。

```
keytool -keystore server.keystore.jks -alias localhost -certreq -file server.csr -storepass xxx
```

“storepass”表示指定密钥库的密码，输入[步骤1.1](#)中设置的keystore密码。

步骤5 使用CA为服务端证书签名。

- 执行以下命令，创建“sans.ext”文件。

```
touch sans.ext
```

- 执行以下命令，编辑“sans.ext”文件，添加以下内容。

```
vim sans.ext
```

添加以下内容。

```
subjectAltName=IP:xxx.xxx.xx.xx,IP:xxx.xxx.xx.xx,IP:xxx.xxx.xx.xx
```

IP地址为Kafka实例连接地址中的IP地址，从[前提条件](#)中获取。

- 执行以下命令，使用CA为服务端证书签名。

```
openssl x509 -req -CA ca.crt -CAkey ca.key -in server.csr -out server.crt -days 3650 -CAcreateserial -
extfile sans.ext
```

返回信息如下：

```
[root@ecs-kafka ~]# openssl x509 -req -CA ca.crt -CAkey ca.key -in server.csr -out server.crt -days
3650 -CAcreateserial -extfile sans.ext
Signature ok
subject=C=xx, ST=js, L=nj, O=test01, OU=test, CN=Tom
Getting CA Private Key
```

步骤6 导入CA证书到服务器keystore。

```
keytool -keystore server.keystore.jks -alias CARoot -import -file ca.crt -storepass xxx -noprompt
```

“storepass”表示指定密钥库的密码，输入[步骤1.1](#)中设置的keystore密码。

返回信息如下：

```
[root@ecs-kafka ~]# keytool -keystore server.keystore.jks -alias CARoot -import -file ca.crt -storepass xxx -
noprompt
Certificate was added to keystore
```

步骤7 导入服务端证书到服务器keystore。

```
keytool -keystore server.keystore.jks -alias localhost -import -file server.crt -storepass xxx -noprompt
```

“storepass”表示指定密钥库的密码，输入[步骤1.1](#)中设置的keystore密码。

返回信息如下：

```
[root@ecs-kafka ~]# keytool -keystore server.keystore.jks -alias localhost -import -file server.crt -storepass xxx -noprompt
Certificate reply was installed in keystore
```

步骤8 通过CA创建一个服务器Truststore证书。

```
keytool -keystore server.truststore.jks -alias CARoot -import -file ca.crt -storepass xxx -noprompt
```

“storepass”表示服务器Truststore证书的密码，根据以下密码要求设置并记录密码，此密码将在后续步骤中使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

返回信息如下：

```
[root@ecs-kafka ~]# keytool -keystore server.truststore.jks -alias CARoot -import -file ca.crt -storepass xxx -noprompt
Certificate was added to keystore
```

步骤9 通过CA创建一个客户端Truststore证书。

```
keytool -keystore client.truststore.jks -alias CARoot -import -file ca.crt -storepass xxx -noprompt
```

“storepass”表示客户端Truststore证书的密码，根据以下密码要求设置并记录密码，此密码将在后续步骤中使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[替换证书对话框](#)中输入密码，系统会自动判定是否为弱密码）。

返回信息如下：

```
[root@ecs-kafka ~]# keytool -keystore client.truststore.jks -alias CARoot -import -file ca.crt -storepass xxx -noprompt
Certificate was added to keystore
```

步骤10 导出证书“server.keystore.jks”、“server.truststore.jks”和“client.truststore.jks”到本地PC机。

“server.keystore.jks”和“server.truststore.jks”在后续步骤“替换证书”时，用来替换“Keystore文件”和“Truststore文件”。请将“client.truststore.jks”存放到客户端指定位置中，并记录其存放路径，此路径将作为客户端连接Kafka实例时，配置文件中“ssl.truststore.location”参数的值。

步骤11 参考[步骤二：替换证书](#)，在Kafka实例控制台替换SSL证书。

表 5-20 替换 SSL 证书参数说明

参数名称	说明
Key密码	输入制作证书中设置的keystore密码。
Keystore密码	输入制作证书中设置的keystore密码。
Keystore文件	导入“server.keystore.jks”证书。
Truststore密码	输入服务端证书的Truststore密码。
Truststore文件	导入“server.truststore.jks”证书。

步骤12 修改客户端配置文件。

替换证书后，需要在客户端的“consumer.properties”和“producer.properties”文件中，分别修改“ssl.truststore.location”和“ssl.truststore.password”参数。

```
security.protocol=SASL_SSL
ssl.truststore.location=/opt/kafka_2.12-2.7.2/config/client.truststore.jks
ssl.truststore.password=axxxb
ssl.endpoint.identification.algorithm=https
```

表 5-21 配置文件参数说明

参数名称	说明
ssl.truststore.location	client.truststore.jks证书的存放路径。
ssl.truststore.password	客户端证书的Truststore密码。
ssl.endpoint.identification.algorithm	证书域名校验开关，设置为“https”，表示开启证书域名校验。

步骤13 验证证书是否生效。

参考[使用客户端连接Kafka（密文接入）](#)，生产和消费消息。如果成功，表示替换的证书已生效。

----结束

5.2.3 获取和使用 Kafka PEM 格式的 SSL 证书

本章节指导您如何获取PEM格式的SSL证书，并使用PEM格式证书访问Kafka实例。

前提条件

实例已开启Kafka SASL_SSL。

获取 PEM 格式的 SSL 证书

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在“连接信息 > SSL证书”后，单击“下载”，将SSL证书下载到本地。

步骤5 解压zip包，获取PEM格式的SSL证书：client.pem。

----结束

使用 PEM 格式证书访问 Kafka 实例

以下示例演示在Java客户端使用PEM格式证书访问Kafka实例。

参考[Java客户端接入示例](#)，连接Kafka实例生产消费消息。其中，生产消息配置文件和消费消息配置文件中的SASL信息修改为如下内容。

```
#SASL认证机制为“PLAIN”时，配置信息如下。
saslm.echanism=PLAIN
saslj.aas.config=org.apache.kafka.common.security.plain.PlainLoginModule required \
username="username" \
password="password";

#SASL认证机制为“SCRAM-SHA-512”时，配置信息如下。
saslm.echanism=SCRAM-SHA-512
saslj.aas.config=org.apache.kafka.common.security.scram.ScramLoginModule required \
username="username" \
password="password";

#设置Kafka安全协议。
security.protocol=SASL_SSL
#ssl.truststore.location为SSL证书的存放路径，如下代码以Windows系统路径格式举例，您在使用时请根据实际运行环境修改路径格式。
ssl.truststore.location=E:\temp\client.pem
#ssl.truststore.password为服务器证书密码，使用PEM格式证书访问Kafka实例，无需配置
ssl.truststore.password参数。
#ssl.truststore.password=dms@kafka
#ssl.endpoint.identification.algorithm为证书域名校验开关，为空则表示关闭，这里需要保持关闭状态，必须设置为空。
ssl.endpoint.identification.algorithm=
#添加ssl.truststore.type参数，指定客户端使用的证书类型是PEM。
ssl.truststore.type=PEM
```

相关文档

[不同实例中，使用的SSL证书是否一样？](#)

5.2.4 配置 Kafka 的 SSL 双向认证

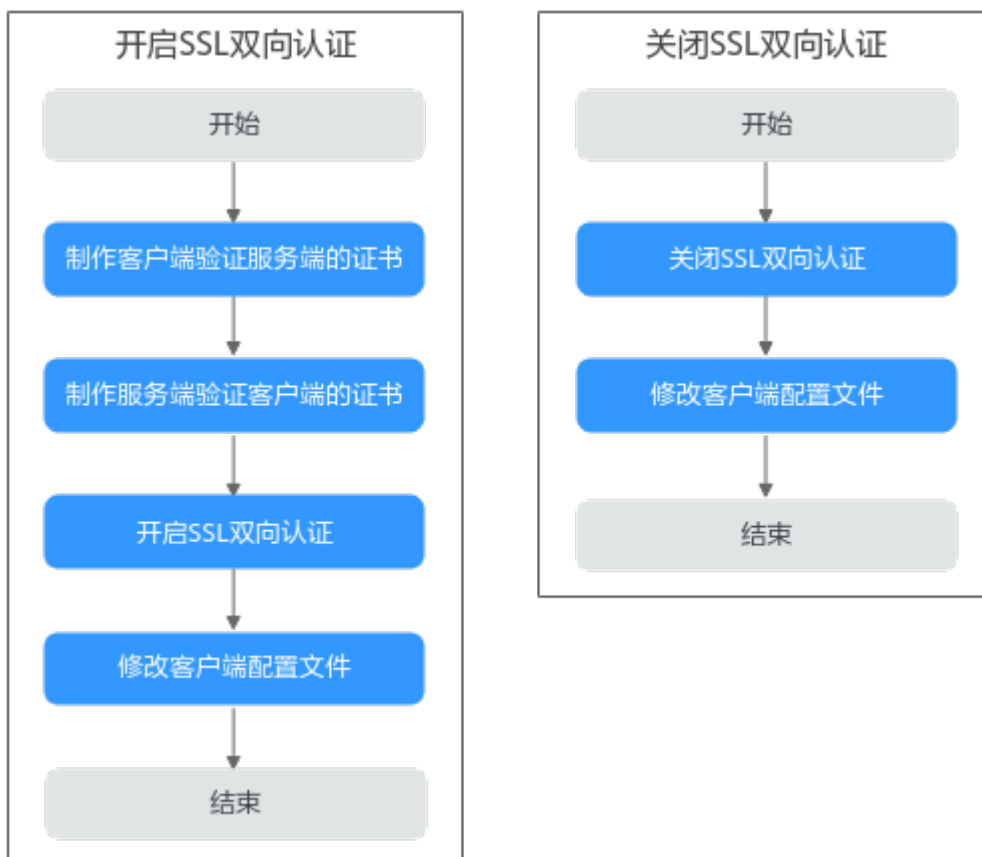
SSL双向认证是指在通信认证时，将同时验证客户端和服务端的证书，以确保通信中涉及的双方都是受信任的。

开启SSL双向认证，主要用于对安全性要求较高的场景。

SSL双向认证功能为受限使用，需要提交工单申请开通。

配置SSL双向认证的流程如[图5-22](#)所示。

图 5-22 配置 SSL 双向认证流程图



约束与限制

开启/关闭SSL双向认证将重启实例，请谨慎操作。

前提条件

- 已准备一台Linux系统的服务器，并安装**Java Development Kit 1.8.111或以上版本**，配置JAVA_HOME与PATH环境变量。
- 实例已开启Kafka SASL_SSL。

步骤一：制作客户端验证服务端的证书

步骤1 登录Linux系统的服务器，执行以下命令，生成server.keystore.jks证书的孵化器。

```
keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA
```

按照提示信息输入keystore密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]:;',<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在**开启SSL双向认证对话框**中输入密码，系统会自动判定是否为弱密码）。

按照提示信息输入证书拥有者的信息，包含姓名、单位、组织名称、所在城市、所在省、所在国家，示例如下。

```
[root@ecs-kafka ~]# keytool -genkey -keystore server.keystore.jks -alias localhost -validity 3650 -keyalg RSA
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Tom
What is the name of your organizational unit?
[Unknown]: test
What is the name of your organization?
[Unknown]: test01
What is the name of your City or Locality?
[Unknown]: nj
What is the name of your State or Province?
[Unknown]: js
What is the two-letter country code for this unit?
[Unknown]: xx
Is CN=Tom, OU=test, O=test01, L=nj, ST=js, C=xx correct?
[no]: y
```

步骤2 执行以下命令，创建CA的认证证书。

```
openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
```

按照提示信息输入PEM密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：4~1024个字符。

按照提示信息输入证书拥有者的信息，包含所在国家、所在省、所在城市、组织名称、单位、姓名、邮箱，示例如下。

```
[root@ecs-kafka ~]# openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'ca-key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:xx
State or Province Name (full name) []:js
Locality Name (eg, city) [Default City]:nj
Organization Name (eg, company) [Default Company Ltd]:test01
Organizational Unit Name (eg, section) []:test
Common Name (eg, your name or your server's hostname) []:Tom
Email Address []:xx
[root@ecs-kafka ~]#
```

步骤3 执行以下命令，从**步骤1**中生成的server.keystore.jks中导出证书，并命名为“server.crt”。

```
keytool -keystore server.keystore.jks -alias localhost -certreq -file server.crt
```

按照提示信息输入**步骤1**中记录的keystore密码。

步骤4 执行以下命令，使用CA的私钥对server.crt进行签名，并为签名后的证书命名为“server-signed.crt”。

```
openssl x509 -req -CA ca-cert -CAkey ca-key -in server.crt -out server-signed.crt -days 3650 -CAcreateserial
```

按照提示信息输入**步骤2**中记录的PEM密码。

步骤5 执行以下命令，将CA证书和“server-signed.crt”导入密钥仓库。

```
keytool -keystore server.keystore.jks -alias CARoot -import -file ca-cert
keytool -keystore server.keystore.jks -alias localhost -import -file server-signed.crt
```

按照提示信息输入**步骤1**中记录的keystore密码。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤6 执行以下命令，使客户端信任服务端证书。

```
keytool -keystore client.truststore.jks -alias CARoot -import -file ca-cert
```

按照提示信息输入client.truststore.jks的密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[开启SSL双向认证对话框](#)中输入密码，系统会自动判定是否为弱密码）。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤7 导出“client.truststore.jks”和“server.keystore.jks”证书到本地PC机。

“server.keystore.jks”在后续步骤“开启SSL双向认证”时，用来替换“Keystore文件”。“client.truststore.jks”需要存放到客户端中，并记录存放路径，此路径用于客户端连接Kafka实例时，配置文件中“ssl.truststore.location”参数的值。

----结束

步骤二：制作服务端验证客户端的证书

步骤1 登录Linux系统的服务器，执行以下命令，生成client.keystore.jks证书的孵化器。

```
keytool -genkey -keystore client.keystore.jks -alias localhost -validity 3650 -keyalg RSA
```

按照提示信息输入keystore密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[]{}:;'",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[开启SSL双向认证对话框](#)中输入密码，系统会自动判定是否为弱密码）。

按照提示信息输入证书拥有者的信息，包含姓名、单位、组织名称、所在城市、所在省、所在国家，示例如下。

```
[root@ecs-kafka ~]# keytool -genkey -keystore client.keystore.jks -alias localhost -validity 3650 -keyalg RSA
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Tom
What is the name of your organizational unit?
[Unknown]: test
```

```
What is the name of your organization?
[Unknown]: test01
What is the name of your City or Locality?
[Unknown]: nj
What is the name of your State or Province?
[Unknown]: js
What is the two-letter country code for this unit?
[Unknown]: xx
Is CN=Tom, OU=test, O=test01, L=nj, ST=js, C=xx correct?
[no]: y
```

步骤2 执行以下命令，创建CA的认证证书。

```
openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
```

按照提示信息输入PEM密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：4~1024个字符。

按照提示信息输入证书拥有者的信息，包含所在国家、所在省、所在城市、组织名称、单位、姓名、邮箱，示例如下。

```
[root@ecs-kafka ~]# openssl req -new -x509 -keyout ca-key -out ca-cert -days 3650
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'ca-key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:xx
State or Province Name (full name) []:js
Locality Name (eg, city) [Default City]:nj
Organization Name (eg, company) [Default Company Ltd]:test01
Organizational Unit Name (eg, section) []:test
Common Name (eg, your name or your server's hostname) []:Tom
Email Address []:xx
[root@ecs-kafka ~]#
```

步骤3 执行以下命令，从**步骤1**中生成的client.keystore.jks中导出证书，并命名为“client.crt”。

```
keytool -keystore client.keystore.jks -alias localhost -certreq -file client.crt
```

按照提示信息输入**步骤1**中记录的keystore密码。

步骤4 执行以下命令，使用CA的私钥对client.crt进行签名，并为签名后的证书命名为“client-signed.crt”。

```
openssl x509 -req -CA ca-cert -CAkey ca-key -in client.crt -out client-signed.crt -days 3650 -CAcreateserial
```

按照提示信息输入**步骤2**中记录的PEM密码。

步骤5 执行以下命令，将CA证书和“client-signed.crt”导入密钥仓库。

```
keytool -keystore client.keystore.jks -alias CARoot -import -file ca-cert
keytool -keystore client.keystore.jks -alias localhost -import -file client-signed.crt
```

按照提示信息输入**步骤1**中记录的keystore密码。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤6 执行以下命令，使服务端信任客户端证书。

```
keytool -keystore server.truststore.jks -alias CARoot -import -file ca-cert
```

按照提示信息输入server.truststore.jks的密码，并记录此密码，此密码在后续步骤中会使用。

密码需要满足如下要求：

- 8~32个字符。
- 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\|[{ }:",<.>/?和空格，并且不能以-开头。
- 非弱密码（如果不能确定哪些是弱密码，请在[开启SSL双向认证对话框](#)中输入密码，系统会自动判定是否为弱密码）。

出现如下提示信息时，输入“y”。

```
Trust this certificate?
```

步骤7 导出“server.truststore.jks”和“client.keystore.jks”证书到本地PC机。

“server.truststore.jks”在后续步骤“开启SSL双向认证”时，用来替换“Truststore文件”。“client.keystore.jks”需要存放到客户端中，并记录存放路径，此路径用于客户端连接Kafka实例时，配置文件中“ssl.keystore.location”参数的值。

----结束

步骤三：开启 SSL 双向认证

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击待开启SSL双向认证的实例名称，进入实例详情页面。

步骤4 在“连接信息 > SSL双向认证”后，单击 ，开启SSL双向认证。

步骤5 在弹出的“SSL双向认证”对话框中，参考[表5-22](#)，设置开启SSL双向认证的参数。

开启SSL双向认证将重启实例，请谨慎操作。

图 5-23 开启 SSL 双向认证

SSL 双向认证

1. 开启或关闭 SSL 双向认证将重启实例。

2. 开启双向认证后，协议将变为SSL。SSL协议不支持ACL权限配置。

Key 密码

Keystore 密码

Keystore 文件

添加文件

只支持 jks 文件且大小不超过 100 KB

Truststore 密码

Truststore 文件

添加文件

只支持 jks 文件且大小不超过 100 KB

表 5-22 开启 SSL 双向认证参数说明

参数名称	说明
Key密码	输入server.keystore.jks的密码
Keystore密码	输入server.keystore.jks的密码
Keystore文件	导入“server.keystore.jks”证书
Truststore密码	输入server.truststore.jks的密码
Truststore文件	导入“server.truststore.jks”证书

步骤6 单击“确定”，完成开启SSL双向认证。

----结束

步骤四：修改客户端配置文件

开启SSL双向认证后，需要在客户端的“consumer.properties”和“producer.properties”文件中，分别修改服务端证书配置，并增加客户端证书配置。

```
security.protocol=SSL
ssl.truststore.location=/opt/kafka_2.12-2.7.2/config/client.truststore.jks
ssl.truststore.password=axxb
ssl.endpoint.identification.algorithm=
#增加以下的客户端证书配置
```

```
ssl.keystore.location=/var/private/ssl/kafka/client.keystore.jks
ssl.keystore.password=txxx3
ssl.key.password=txxx3
```

表 5-23 配置文件参数说明

参数名称	说明
security.protocol	证书协议类型，开启SSL双向认证时，必须设置为SSL。
ssl.truststore.location	client.truststore.jks证书的存放路径。
ssl.truststore.password	client.truststore.jks的密码。
ssl.endpoint.identification.algorithm	证书域名校验开关，为空则表示关闭。这里需要保持关闭状态，必须设置为空。
ssl.keystore.location	client.keystore.jks证书的存放路径。
ssl.keystore.password	client.keystore.jks的密码。
ssl.key.password	client.keystore.jks的密码。

步骤五：验证证书是否生效

参考[使用客户端连接Kafka（密文接入）](#)，生产和消费消息。如果成功，表示替换的证书已生效。

关闭 SSL 双向认证

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击待关闭SSL双向认证的实例名称，进入实例详情页面。
- 步骤4 在“连接信息 > SSL双向认证”后，单击 ，关闭SSL双向认证。

 **注意**

关闭SSL双向认证将重启实例，请谨慎操作。

- 步骤5 关闭SSL双向认证后，需要在客户端的“consumer.properties”和“producer.properties”文件中，分别修改服务端证书协议类型，并删除客户端证书配置。

```
security.protocol=SASL_SSL
ssl.truststore.location=/opt/kafka_2.12-2.7.2/config/client.truststore.jks
ssl.truststore.password=axxxb
ssl.endpoint.identification.algorithm=
#删除以下的客户端证书配置
ssl.keystore.location=/var/private/ssl/kafka.client.keystore.jks
ssl.keystore.password=txxx3
ssl.key.password=txxx3
```

security.protocol配置证书协议类型，关闭SSL双向认证时，必须设置为SASL_SSL。
“ssl.truststore.location”、“ssl.truststore.password”和
“ssl.endpoint.identification.algorithm”无需修改。

----结束

5.2.5 配置 Kafka ACL 用户

Kafka实例开启密文接入后，支持对Topic进行ACL（Access Control List）权限管理，您可以在Topic中为不同的用户设置不同的生产消费消息的权限，以达到用户之间的权限隔离。

本章节主要介绍Kafka实例开启密文接入后，如何创建用户、重置用户密码、修改用户信息和删除用户。创建用户后，对Topic进行用户授权的操作，请参见[设置Kafka Topic权限](#)。

约束与限制

- 单机实例不支持创建用户、重置用户密码、修改用户信息和删除用户。
- 首次开启密文接入时设置的用户无法删除。
- 重置用户密码会导致业务中断，您需要及时在客户端配置文件或代码中修改用户密码。
- 一个Kafka实例最多创建的用户数量在控制台存在两种限制，一种为20个，另一种为500个，具体以控制台为准。

前提条件

- Kafka实例已开启密文接入。
- Kafka实例的“状态”为“运行中”。

创建用户

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > 用户管理”，进入用户列表页面。
- 步骤5 在页面左上方，单击“创建用户”。
- 步骤6 在弹出的“创建用户”对话框中，参考[表5-24](#)，设置用户信息。

表 5-24 创建用户参数说明

参数	说明
用户名	连接Kafka实例的用户名，用户名支持自定义，但需要符合命名规则：长度为4~64个字符，由英文字母开头，只能由英文字母、数字、中划线、下划线组成。

参数	说明
密码	连接Kafka实例的密码，密码需要符合以下规则： - 长度为8~32个字符。 - 至少包含以下字符中的3种：大写字母、小写字母、数字、特殊字符`~!@#\$%^&*()-_+=\ [{ }];:","<.>?`和空格，并且不能以-开头。 - 不能与用户名或倒序的用户名相同。
描述	用户的描述信息，长度为0~200个字符。

步骤7 单击“确定”，完成用户的创建。

在用户列表页面，查看新创建的用户。

用户创建成功后，参考[设置Kafka Topic权限](#)为用户授权。

----结束

重置用户密码

在控制台创建用户的方法有两种，两种方法对应的重置密码的步骤有区别。

- 初始用户：在首次开启密文接入时设置的用户，如果忘记了密码，请参考重置用户密码（初始用户），重置密码。
- 非初始用户：在“实例管理 > 用户管理”页面创建的用户，如果忘记了密码，请参考重置用户密码（非初始用户），重置密码。

重置用户密码（初始用户）

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 通过以下任意一种方法，重置初始用户的密码。

- 在需要重置密码的Kafka实例右侧，单击“更多 > 重置Kafka密码”，弹出“重置Kafka密码”对话框。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 重置Kafka密码”，弹出“重置Kafka密码”对话框。
- 单击Kafka实例名称，进入实例详情页面。在“概览”页面，在“连接信息 > 用户名”后，单击“重置密码”，弹出“重置Kafka密码”对话框。
- 单击Kafka实例名称，进入实例详情页面。在“实例管理 > 用户管理”页面，在待重置密码的用户所在行，单击“重置密码”，弹出“重置密码”对话框。

步骤4 输入“新密码”，单击“确定”，完成密码重置。

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

只有所有代理都重置密码成功，系统才会提示重置密码成功，否则会提示重置失败。

----结束

重置用户密码（非初始用户）

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击待重置用户密码的Kafka实例名称，进入实例详情页面。

步骤4 在“实例管理 > 用户管理”页面，在待重置密码的用户所在行，单击“重置密码”。

步骤5 输入“新密码”，单击“确定”，完成密码重置。

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

说明

只有所有代理都重置密码成功，系统才会提示重置密码成功，否则会提示重置失败。

----结束

修改用户信息

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > 用户管理”，进入用户列表页面。

步骤5 在待修改描述信息用户所在行，单击“编辑”，弹出“编辑”对话框。

步骤6 修改描述信息，单击“确定”。

修改完成后，在用户列表页面的“描述”列查看修改后的描述信息。

----结束

删除用户

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > 用户管理”，进入用户列表页面。

步骤5 通过以下任意一种方法，删除用户。

- 在待删除的用户所在行，单击“删除”。

- 勾选用户名左侧的方框，可选一个或多个，单击信息栏左上侧的“删除”。

步骤6 在弹出的“删除用户”对话框中，单击“确定”，完成用户的删除。

已删除的用户不再显示在用户列表中，表明用户已成功删除。

----结束

导出用户列表

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > 用户管理”，进入用户列表页面。

步骤5 通过以下任意一种方法，导出用户列表。

- 勾选待导出的用户，单击“导出 > 导出已选中数据到XLSX”，导出指定用户列表。
- 单击“导出 > 导出全部数据到XLSX”，导出全部用户列表。

请注意：初始用户无法被选中，如果要导出初始用户，请使用“导出全部数据到XLSX”。

----结束

相关文档

- 如果您希望通过API创建用户，请参考[创建用户](#)。
- 如果您希望通过API重置用户密码，请参考[重置用户密码](#)。
- 如果您希望授予用户发布/订阅Topic的权限，请参考[设置Kafka Topic权限](#)。

5.3 配置 Kafka 客户端

5.3.1 Kafka 客户端参数配置建议

Kafka客户端的配置参数很多，以下提供Producer和Consumer几个常用参数配置。不同版本的Kafka客户端参数名称可能不同，以下配置参数适用于1.1.0及以上版本。其他参数和版本配置，请参考[Kafka配置](#)。

表 5-25 Producer 参数

参数	默认值	推荐值	说明
acks	1	高可靠：all或者-1 高吞吐：1	收到Server端确认信号个数，表示producer需要收到多少个这样的确认信号，算消息发送成功。acks参数代表了数据备份的可用性。常用选项： acks=0：表示producer不需要等待任何确认收到的信息，副本将立即加到socket buffer并认为已经发送。没有任何保障可以保证此种情况下server已经成功接收数据，同时重试配置不会发生作用（因为客户端不知道是否失败）回馈的offset会总是设置为-1。 acks=1：这意味着至少要等待leader已经成功将数据写入本地log，但是并没有等待所有follower是否成功写入。如果follower没有成功备份数据，而此时leader又无法提供服务，则消息会丢失。 acks=all或者-1：这意味着leader需要等待ISR中所有备份都成功写入日志。只要任何一个备份存活，数据都不会丢失。min.insync.replicas指定必须确认写入才能被认为成功的副本的最小数量。

参数	默认值	推荐值	说明
retries	- 客户端版本为 2.1.0及以上：Integer.MAX_VALUE - 客户端版本为 2.1.0以下：0	结合实际业务调整	客户端发送消息的重试次数。值大于0时，这些数据发送失败后，客户端会重新发送。 注意，这些重试与客户端接收到发送错误时的重试没有什么不同。允许重试将潜在的改变数据的顺序，如果这两个消息记录都是发送到同一个 partition，则第一个消息失败第二个发送成功，则第二条消息会比第一条消息出现要早。 针对网络闪断场景，生产者建议配置重试能力，具体如下： - 客户端版本为2.1.0及以上：推荐重试次数retries和总超时时间 delivery.timeout.ms均保持默认值，重试间隔 retry.backoff.ms=1000。 - 客户端版本为2.1.0以下：推荐重试次数retries=10，重试间隔 retry.backoff.ms=1000。
request.timeout.ms	30000	结合实际业务调整	设置一个请求最大等待时间（单位为ms），超过这个时间则会抛Timeout异常。 超时时间如果设置大一些，如 127000（127秒），高并发的场景中，能减少发送失败的情况。
block.on.buffer.full	TRUE	TRUE	TRUE表示当我们内存用尽时，停止接收新消息记录或者抛出错误。 默认情况下，这个设置为 TRUE。然而某些阻塞可能不值得期待，因此立即抛出错误更好。如果设置为false，则 producer抛出一个异常错误：BufferExhaustedException

参数	默认值	推荐值	说明
batch.size	16384	262144	默认的批量处理消息字节数上限。producer将试图批处理消息记录，以减少请求次数。这将改善client与server之间的性能。不会试图处理大于这个字节数的消息字节数。 发送到brokers的请求将包含多个批量处理，其中会包含对每个partition的一个请求。 较小的批量处理数值比较少用，并且可能降低吞吐量（0则会禁用批量处理）。较大的批量处理数值将会浪费更多内存空间，这样就需要分配特定批量处理数值的内存大小。
buffer.memory	33554432	67108864	producer可以用来缓存数据的内存大小。如果数据产生速度大于向broker发送的速度，producer会阻塞或者抛出异常，以“block.on.buffer.full”来表明。 这项设置将和producer能够使用的总内存相关，但并不是一个硬性的限制，因为不是producer使用的所有内存都是用于缓存。一些额外的内存会用于压缩（如果引入压缩机制），同样还有一些用于维护请求。
enable.idempotence	3.0之前版本：false 3.0及之后版本：true	如果无幂等需求，建议设置为false。	在生产者客户端中开启幂等，然后生产消息，此时您会在消费者客户端或Kafka控制台的“实例管理 > 消息查询”中观察到消息offset不连续的现象。这是因为开启了幂等后，在生产消息时会产生一些元数据控制消息，这些控制消息也会生产到该Topic中，且它们对消费者不可见，从而造成offset不连续的现象。

表 5-26 Consumer 参数

参数	默认值	推荐值	说明
auto.commit.enable	TRUE	FALSE	如果为真，consumer所fetch的消息的offset将会自动的同步到zookeeper。这项提交的offset将在进程无法提供服务时，由新的consumer使用。 约束： 设置为false后，需要先成功消费再提交，这样可以避免消息丢失。
auto.offset.reset	latest	earliest	没有初始化offset或者offset被删除时，可以设置以下值： • earliest：自动复位offset为最早 • latest：自动复位offset为最新 • none：如果没有发现offset，则向消费者抛出异常 • anything else：向消费者抛出异常。 如果将此配置设置为latest，新增分区时，生产者可能会在消费者重置初始偏移量之前开始向新增加的分区发送消息，从而导致部分消息丢失。
connections.max.idle.ms	600000	30000	空连接的超时时间（单位为ms），设置为30000可以在网络异常场景下减少请求卡顿的时间。
max.poll.records	500	需小于max.poll.interval.ms时间内的消费处理能力。	消费者单次从Broker中拉取的最大消息条数。
max.poll.interval.ms	300000	如果在两次poll之间存在复杂、耗时的逻辑，需要延长该参数值。	消费者两次拉取消息的最大时间间隔，单位为ms。如果两次拉取消息的时间间隔超过该间隔时间，就视为此次消费失败，消费者被踢出消费组，触发Rebalance。
heartbeat.interval.ms	3000	>=3000	消费者和Kafka之间的心跳间隔，单位为ms。

参数	默认值	推荐值	说明
session.timeout.ms	10000	将该参数值设置为 heartbeat.interval.ms 的 3 倍以上。	使用消费组管理 offset 时，消费者与 Broker 的会话超时时间，单位为 ms。
fetch.max.bytes	1000000	max.request.size < message.max.bytes < fetch.max.bytes	消费者单次从 Broker 中拉取消息的最大字节数。

5.3.2 Kafka 客户端使用建议

consumer 使用建议

1. consumer 的 owner 线程需确保不会异常退出，避免客户端无法发起消费请求，阻塞消费。
2. 确保处理完消息后再做消息 commit，避免业务消息处理失败，无法重新拉取处理失败的消息。
3. 通常不建议对每条消息都进行 commit，如果对每条消息都进行了 commit，会导致 OFFSET_COMMIT 请求过多，进而导致 CPU 使用率过高。例如：如果一个消费请求拉取 1000 条消息，每条都 commit，则 commit 请求 TPS 是消费的 1000 倍，消息体越小，这个比例越大。建议隔一定条数或时间，批量 commit，或打开 enable.auto.commit，这样设置会存在一个缺点，即在客户端故障时，可能丢失一部分缓存的消费进度，导致重复消费。请根据业务实际情况，设置批量 commit。
4. consumer 不能频繁加入和退出 group，频繁加入和退出，会导致 consumer 频繁做 rebalance，阻塞消费。
5. 同一消费组内 consumer 数量不能超过该消费组订阅的分区总数，否则会有 consumer 拉取不到消息。
6. consumer 需周期 poll，维持和 server 的心跳，避免心跳超时，导致 consumer 频繁加入和退出，阻塞消费。
7. consumer 拉取的消息本地缓存应有大小限制，避免 OOM（Out of Memory）。
8. consumer session 设置为 30 秒，session.timeout.ms=30000。
9. Kafka 不能保证消费重复的消息，业务侧需保证消息处理的幂等性。
10. 消费线程退出要调用 consumer 的 close 方法，避免同一个组的其他消费者阻塞 session.timeout.ms 的时间。
11. 消费组名称开头不使用特殊字符（如 #），使用特殊字符可能会导致云监控无法展示此消费组的监控数据。
12. 在消费逻辑中处理 AuthorizationException，设置重试消费。如在 SpringKafka 中，可以在配置文件中增加以下配置。
设置授权异常重试间隔为 10 秒
spring.kafka.listener.authorizationExceptionRetryInterval=10000

producer 使用建议

1. 同步复制客户端需要配合使用：acks=all
2. 配置发送失败重试：retries=10
3. 配置发送重试间隔：retry.backoff.ms=1000
4. 发送优化：对于时延敏感的信息，设置linger.ms=0。对于时延不敏感的信息，设置linger.ms在100~1000之间。
5. 生产端的JVM内存要足够，避免内存不足导致发送阻塞。
6. 时间戳设置为与当地时间一致，避免时间戳为未来时间导致消息无法老化。
7. 尽量复用producer，不要频繁创建producer。当producer开启幂等时（生产者客户端3.0及之后的版本默认开启幂等），生产消息会在服务端创建生产者状态对象，若频繁创建producer，会导致服务端创建大量生产者状态对象后无法及时回收，服务端内存占用飙升，进而导致服务端性能急剧下降。如果不需要使用幂等功能，请将“enable.idempotence”设置为“false”。
8. 捕获AuthorizationException异常，建议在业务侧重试生产消息，可以采取有限重试及退避策略实现自愈。

Topic 使用建议

配置要求：推荐3副本，同步复制，最小同步副本数为2，且同步副本数不能等于Topic副本数，否则宕机1个副本会导致无法生产消息。

创建方式：支持选择是否开启Kafka自动创建Topic的开关。选择开启后，表示向一个未创建的Topic生产或消费消息时，系统会自动创建此Topic。

其他建议

连接数限制：3000

消息大小：不能超过10MB

使用SASL_SSL协议访问Kafka：确保DNS具有反向解析能力，或者在hosts文件配置Kafka所有节点IP和主机名映射，避免Kafka client做反向解析，阻塞连接建立。

磁盘容量申请超过业务量 * 副本数的2倍，即保留磁盘空闲50%左右。

业务进程JVM内存使用确保无频繁FGC，否则会阻塞消息的生产和消费。

5.4 使用客户端连接 Kafka（明文接入）

本章节介绍如何使用开源的Kafka客户端访问关闭SASL的Kafka实例的方法。关闭SASL表示客户端访问Kafka实例时，无需进行认证，数据通过明文传输，性能更好。

约束与限制

2020年7月以及之后购买的实例，Kafka实例的每个代理允许客户端单IP连接的个数默认为1000个，在此之前购买的实例，Kafka实例的每个代理允许客户端单IP连接的个数默认为200个，如果超过了，会出现连接失败问题。您可以通过[修改Kafka实例配置参数](#)来修改单IP的连接数，即修改max.connections.per.ip参数值。

前提条件

- 已连接客户端和Kafka实例之间的网络，具体网络要求请参考[连接Kafka网络要求](#)。
- 已配置正确的安全组。
客户端访问关闭SASL的Kafka实例前，Kafka实例需要配置正确的安全组规则，否则会连接失败。安全组规则的配置请参考[表5-2](#)。
- 已获取连接Kafka实例的地址。
在Kafka控制台的“概览 > 连接信息”中获取实例连接地址。Kafka实例连接地址在Kafka控制台存在两种不同的显示，一种为“内网连接地址” / “公网连接地址”，另一种为“内网明文连接地址” / “公网明文连接地址”，具体以控制台显示为准。

- 如果是使用内网通过同一个VPC访问，Kafka连接地址如下图所示。

图 5-24 使用内网通过同一个 VPC 访问 Kafka 实例的连接地址（内网连接地址）



图 5-25 使用内网通过同一个 VPC 访问 Kafka 实例的连接地址（内网明文连接地址）



- 如果是公网访问，Kafka连接地址如下图所示。

图 5-26 公网访问 Kafka 实例的连接地址（公网连接地址）



图 5-27 公网访问 Kafka 实例的连接地址（公网明文连接地址）



- 如果Kafka实例未开启自动创建Topic功能，在连接实例前，请先[创建Topic](#)，否则会连接失败。
- 已下载对应版本的Kafka命令行工具，确保Kafka实例版本与命令行工具版本相同。
 - [Kafka命令行工具1.1.0版本](#)
 - [Kafka命令行工具2.7.2版本](#)
 - [Kafka命令行工具3.4.0版本](#)
- 已在服务器中安装[Java Development Kit 1.8.111或以上版本](#)，并配置JAVA_HOME与PATH环境变量，具体方法如下：

使用执行用户在用户家目录下修改“.bash_profile”，添加如下行。其中“/opt/java/jdk1.8.0_151”为JDK的安装路径，请根据实际情况修改。

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

执行source .bash_profile命令使修改生效。

命令行模式连接实例

以下操作命令以Linux系统为例进行说明：

步骤1 解压Kafka命令行工具。

进入文件压缩包所在目录，然后执行以下命令解压文件。

```
tar -zxf {kafka_tar}
```

其中，{kafka_tar}表示命令行工具的压缩包名称。

例如：

```
tar -zxf kafka_2.12-2.7.2.tgz
```

步骤2 进入Kafka命令行工具的“/bin”目录下。

注意，Windows系统下需要进入“/bin/windows”目录下。

```
cd {kafka_tar}/bin
```

步骤3 执行如下命令进行生产消息。

```
./kafka-console-producer.sh --broker-list {连接地址} --topic {Topic名称}
```

表 5-27 生产消息参数说明

参数名称	说明
连接地址	Kafka实例的连接地址。 从前提条件中获取连接地址。
Topic名称	Kafka实例下创建的Topic名称。 如果Kafka实例开启了自动创建Topic功能，此参数值可以填写已创建的Topic名称，也可以填写未创建的Topic名称。

本文以公网连接为例，获取的Kafka实例公网连接地址为“10.xx.xx.45:9094,10.xx.xx.127:9094,10.xx.xx.103:9094”。执行完命令后输入内容，按“Enter”发送消息到Kafka实例，输入的每一行内容都将作为一条消息发送到Kafka实例。

```
[root@ecs-kafka bin]# ./kafka-console-producer.sh --broker-list
10.xx.xx.45:9094,10.xx.xx.127:9094,10.xx.xx.103:9094 --topic topic-demo
>Hello
>DMS
>Kafka!
>^C[root@ecs-kafka bin]#
```

如需停止生产使用Ctrl+C命令退出。

步骤4 执行如下命令消费消息。

```
./kafka-console-consumer.sh --bootstrap-server {连接地址} --topic {Topic名称} --group {消费组名称} --from-
beginning
```

表 5-28 消费消息参数说明

参数名称	说明
连接地址	Kafka实例的连接地址。 从 前提条件 中获取连接地址。
Topic名称	Kafka实例下创建的Topic名称。
消费组名称	根据您的业务需求，设定消费组名称。消费组名称开头包含特殊字符，例如#号“#”时，监控数据无法展示。

消费消息示例如下：

```
[root@ecs-kafka bin]# ./kafka-console-consumer.sh --bootstrap-server
10.xx.xx.45:9094,10.xx.xx.127:9094,10.xx.xx.103:9094 --topic topic-demo --group order-test --from-beginning
Kafka!
DMS
Hello
^CProcessed a total of 3 messages
[root@ecs-kafka bin]#
```

如需停止消费使用Ctrl+C命令退出。

----结束

相关文档

- [Kafka连接问题排查和解决](#)
- [Kafka实例的内网连接地址可以修改吗？](#)

5.5 使用客户端连接 Kafka（密文接入）

本章节介绍如何使用开源的Kafka客户端访问开启密文接入的Kafka实例的方法。密文接入表示客户端访问Kafka实例时，需要进行SASL认证。如果Kafka安全协议为“SASL_SSL”，客户端和Kafka实例进行通信时，数据使用加密传输，安全性更高。

约束与限制

- 由于安全问题，2021年3月20日前以及当天创建的实例，支持的加密套件为TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256，TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256和TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256。2021年3月20日后创建的实例，支持的加密套件为TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256。
- 2020年7月以及之后购买的实例，Kafka实例的每个代理允许客户端单IP连接的个数默认为1000个，在此之前购买的实例，Kafka实例的每个代理允许客户端单IP连接的个数默认为200个，如果超过了，会出现连接失败问题。您可以通过[修改Kafka实例配置参数](#)来修改单IP的连接数，即修改max.connections.per.ip参数值。

前提条件

- 已连接客户端和Kafka实例之间的网络，具体网络要求请参考[连接Kafka网络要求](#)。

- 已配置正确的安全组。
客户端访问开启密文接入的Kafka实例时，实例需要配置正确的安全组规则，否则会连接失败。安全组规则的配置请参考[表5-2](#)。
- 已获取连接Kafka实例的地址。
在Kafka控制台的“概览 > 连接信息”中获取实例连接地址。Kafka实例连接地址在Kafka控制台存在两种不同的显示，一种为“内网连接地址” / “公网连接地址”，另一种为“内网密文连接地址” / “公网密文连接地址”，具体以控制台显示为准。
 - 如果是使用内网通过同一个VPC访问，Kafka连接地址如下图所示。

图 5-28 使用内网通过同一个 VPC 访问 Kafka 实例的连接地址（内网连接地址）

内网连接地址 IPv4 192.168.0.239:9093,192.168.0.182:9093,192.168.0.57:9093

图 5-29 使用内网通过同一个 VPC 访问 Kafka 实例的连接地址（内网密文连接地址）

内网密文连接地址 IPv4 192.168.0.30:9093,192.16...

- 如果是公网访问，Kafka连接地址如下图所示。

图 5-30 公网访问 Kafka 实例的连接地址（公网连接地址）

公网连接地址 139.145:9095,122.150:9095,119.229:9095

图 5-31 公网访问 Kafka 实例的连接地址（公网密文连接地址）

公网密文连接地址 IPv4 52.2.64:9095,52.1...

- 已获取开启的SASL认证机制。
在Kafka实例详情页的“连接信息”区域，查看“开启的SASL认证机制”。很久前创建的Kafka实例在详情页如果未显示“开启的SASL认证机制”，默认使用PLAIN机制。

图 5-32 开启的 SASL 认证机制

开启的SASL认证机制 SCRAM-SHA-512

- 已获取启用的安全协议。
在Kafka实例详情页的“连接信息”区域，查看“启用的安全协议”。很久前创建的Kafka实例在详情页如果未显示“启用的安全协议”，默认使用SASL_SSL协议。
- 如果Kafka实例未开启自动创建Topic功能，在连接实例前，请先[创建Topic](#)，否则会连接失败。

- 已下载client.jks证书。如果没有，在控制台单击Kafka实例名称，进入实例详情页面，在“连接信息 > SSL证书”所在行，单击“下载”。下载压缩包后解压，获取压缩包中的客户端证书文件：client.jks。如果您已经用自制的SSL证书替换了Kafka提供的证书，请提前准备好您自制的SSL证书。
- 已下载对应版本的Kafka命令行工具，确保Kafka实例版本与命令行工具版本相同。
 - [Kafka命令行工具1.1.0版本](#)
 - [Kafka命令行工具2.7.2版本](#)
 - [Kafka命令行工具3.4.0版本](#)
- 已在服务器中安装[Java Development Kit 1.8.111或以上版本](#)，并配置JAVA_HOME与PATH环境变量，具体方法如下：
使用执行用户在用户家目录下修改“.bash_profile”，添加如下行。其中“/opt/java/jdk1.8.0_151”为JDK的安装路径，请根据实际情况修改。

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

执行source .bash_profile命令使修改生效。

命令行模式连接实例

以下操作命令以Linux系统为例进行说明。

- 步骤1** 在客户端所在主机的“/etc/hosts”文件中配置host和IP的映射关系，以便客户端能够快速解析实例的Broker。
1. 执行以下命令，编辑“/etc/hosts”文件。

```
vim /etc/hosts
```
 2. 在键盘上按“i”，在“/etc/hosts”文件中增加host和IP的映射关系。

```
ip1 主机名称1
ip2 主机名称2
ip3 主机名称3
```

表 5-29 “/etc/hosts”文件参数说明

参数名称	说明
IP地址	Kafka实例的连接地址，从 前提条件 中获取。
主机名称	每个实例主机的名称。 主机的名称由您自行设置，但不能重复。

- 例如：
- ```
10.154.48.120 server01
10.154.48.121 server02
10.154.48.122 server03
```
3. 按“Esc”，然后输入以下命令，按“Enter”，保存并退出“/etc/hosts”文件。

```
:wq
```

- 步骤2** 解压Kafka命令行工具。

进入文件压缩包所在目录，然后执行以下命令解压文件。

```
tar -zxf {kafka_tar}
```

其中，`{kafka_tar}`表示命令行工具的压缩包名称。

例如：

```
tar -zxf kafka_2.12-2.7.2.tgz
```

**步骤3** 修改Kafka命令行工具配置文件。

配置文件包括生产者配置文件“`producer.properties`”和消费者配置文件“`consumer.properties`”，需要分别修改。

**修改“`producer.properties`”文件：**

1. 进入Kafka命令行工具的“`/config`”目录。  
`cd {kafka_tar}/config`
2. 编辑“`producer.properties`”文件。  
`vim producer.properties`
3. 在键盘上按“`i`”，然后在“`producer.properties`”文件中添加如下内容。

a. Kafka实例包含两种认证机制，根据前提条件中获取的**SASL认证机制**，将对**应认证机制的内容添加到配置文件中**。如果SCRAM-SHA-512和PLAIN都开启了，根据实际情况选择其中任意一种配置连接。

■ **PLAIN机制：**

`sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule required \`  
`username="*****" \`  
`password="*****";`  
`sasl.mechanism=PLAIN`

表 5-30 SASL 认证机制参数说明

| 参数名称     | 说明                             |
|----------|--------------------------------|
| username | 首次开启密文接入时设置的用户名，或者创建用户时设置的用户名。 |
| password | 首次开启密文接入时设置的密码，或者创建用户时设置的密码。   |

■ **SCRAM-SHA-512机制：**

`sasl.jaas.config=org.apache.kafka.common.security.scram.ScramLoginModule required \`  
`username="*****" \`  
`password="*****";`  
`sasl.mechanism=SCRAM-SHA-512`

表 5-31 SASL 认证机制参数说明

| 参数名称     | 说明                             |
|----------|--------------------------------|
| username | 首次开启密文接入时设置的用户名，或者创建用户时设置的用户名。 |
| password | 首次开启密文接入时设置的密码，或者创建用户时设置的密码。   |

- b. Kafka实例包含两种安全协议，根据前提条件中获取的**安全协议**，将对**应安全协议的内容添加到配置文件中**。

- **SASL\_SSL:**  
security.protocol=SASL\_SSL  
ssl.truststore.location={ssl\_truststore\_path}  
ssl.truststore.password=dms@kafka  
ssl.endpoint.identification.algorithm=

表 5-32 SASL\_SSL 参数说明

| 参数名称                                  | 说明                                                                                    |
|---------------------------------------|---------------------------------------------------------------------------------------|
| ssl.truststore.location               | client.jks证书的存放路径。<br>注意，Windows系统下证书路径中也必须使用“/”，不能使用Windows系统中复制路径时的“\”，否则客户端获取证书失败。 |
| ssl.truststore.password               | Kafka客户端证书密码。<br>如果使用Kafka控制台提供的SSL证书，默认为dms@kafka，不可更改。如果使用您自制的客户端证书，请根据实际情况配置。      |
| ssl.endpoint.identification.algorithm | 证书域名校验开关，为空则表示关闭。这里需要保持关闭状态，必须设置为空。                                                   |

- **SASL\_PLAINTEXT:**  
security.protocol=SASL\_PLAINTEXT

4. 按“Esc”，然后输入以下命令，按“Enter”，保存并退出“producer.properties”文件。  
:wq

修改“consumer.properties”文件:

1. 编辑“consumer.properties”文件。  
vim consumer.properties
2. 在键盘上按“i”，然后在“consumer.properties”文件中添加如下内容。

a. Kafka实例包含两种认证机制，根据前提条件中获取的SASL认证机制，将对应认证机制的内容添加到配置文件中。如果SCRAM-SHA-512和PLAIN都开启了，根据实际情况选择其中任何一种配置连接。

- **PLAIN机制:**  
sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule required \  
username="\*\*\*\*\*" \  
password="\*\*\*\*\*";  
sasl.mechanism=PLAIN

表 5-33 SASL 认证机制参数说明

| 参数名称     | 说明                             |
|----------|--------------------------------|
| username | 首次开启密文接入时设置的用户名，或者创建用户时设置的用户名。 |

| 参数名称     | 说明                           |
|----------|------------------------------|
| password | 首次开启密文接入时设置的密码，或者创建用户时设置的密码。 |

■ **SCRAM-SHA-512机制：**

```
sasl.jaas.config=org.apache.kafka.common.security.scram.ScramLoginModule required \
username="*****" \
password="*****",
sasl.mechanism=SCRAM-SHA-512
```

表 5-34 SASL 认证机制参数说明

| 参数名称     | 说明                             |
|----------|--------------------------------|
| username | 首次开启密文接入时设置的用户名，或者创建用户时设置的用户名。 |
| password | 首次开启密文接入时设置的密码，或者创建用户时设置的密码。   |

- b. Kafka实例包含两种安全协议，根据前提条件中获取的[安全协议](#)，将对应安全协议的内容添加到配置文件中。

■ **SASL\_SSL：**

```
security.protocol=SASL_SSL
ssl.truststore.location={ssl_truststore_path}
ssl.truststore.password=dms@kafka
ssl.endpoint.identification.algorithm=
```

表 5-35 SASL\_SSL 参数说明

| 参数名称                                  | 说明                                                                                    |
|---------------------------------------|---------------------------------------------------------------------------------------|
| ssl.truststore.location               | client.jks证书的存放路径。<br>注意，Windows系统下证书路径中也必须使用“/”，不能使用Windows系统中复制路径时的“\”，否则客户端获取证书失败。 |
| ssl.truststore.password               | Kafka客户端证书密码。<br>如果使用Kafka控制台提供的SSL证书，默认为dms@kafka，不可更改。如果使用您自制的客户端证书，请根据实际情况配置。      |
| ssl.endpoint.identification.algorithm | 证书域名校验开关，为空则表示关闭。这里需要保持关闭状态，必须设置为空。                                                   |

■ **SASL\_PLAINTEXT：**

```
security.protocol=SASL_PLAINTEXT
```

3. 按“Esc”，然后输入以下命令，按“Enter”，保存并退出“consumer.properties”文件。

```
:wq
```

**步骤4** 进入Kafka命令行工具的“/bin”目录下。

注意，Windows系统下需要进入“/bin/windows”目录下。

```
cd ../bin
```

**步骤5** 执行如下命令进行生产消息。

```
./kafka-console-producer.sh --broker-list {连接地址} --topic {Topic名称} --producer.config ../config/producer.properties
```

表 5-36 生产消息参数说明

| 参数名称    | 说明                                                                                   |
|---------|--------------------------------------------------------------------------------------|
| 连接地址    | Kafka实例的连接地址。<br>从 <a href="#">前提条件</a> 中获取连接地址。                                     |
| Topic名称 | Kafka实例下创建的Topic名称。<br>如果Kafka实例开启了自动创建Topic功能，此参数值可以填写已创建的Topic名称，也可以填写未创建的Topic名称。 |

本文以公网访问为例，Kafka实例连接地址为  
“10.xx.xx.45:9095,10.xx.xx.127:9095,10.xx.xx.103:9095”。

执行完命令后，输入需要生产的消息内容，按“Enter”发送消息到Kafka实例，输入的每一行内容都将作为一条消息发送到Kafka实例。

```
[root@ecs-kafka bin]#./kafka-console-producer.sh --broker-list
10.xx.xx.45:9095,10.xx.xx.127:9095,10.xx.xx.103:9095 --topic topic-demo --producer.config ../config/
producer.properties
>Hello
>DMS
>Kafka!
>^C[root@ecs-kafka bin]#
```

如需停止生产使用Ctrl+C命令退出。

**步骤6** 执行如下命令消费消息。

```
./kafka-console-consumer.sh --bootstrap-server {连接地址} --topic {Topic名称} --group {消费组名称} --from-
beginning --consumer.config ../config/consumer.properties
```

表 5-37 消费消息参数说明

| 参数名称    | 说明                                               |
|---------|--------------------------------------------------|
| 连接地址    | Kafka实例的连接地址。<br>从 <a href="#">前提条件</a> 中获取连接地址。 |
| Topic名称 | Kafka实例下创建的Topic名称。                              |

| 参数名称  | 说明                                                                                                                                                                                      |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 消费组名称 | 根据您的业务需求，设定消费组名称。消费组名称开头包含特殊字符，例如#号“#”时，监控数据无法展示。<br><b>警告</b><br>配置文件（即consumer.properties）中已经指定了消费组名称，请确保命令中的消费组名称与配置文件中的相同，否则可能消费失败。<br>查看并修改配置文件中消费组名称的方法参考 <a href="#">步骤6.1</a> 。 |

查看并修改配置文件中消费组名称的方法如下：

1. 进入Kafka命令行工具的“/config”目录下。

cd ../config
2. 编辑“consumer.properties”文件。

vim consumer.properties
3. 在键盘上按“i”，查看并修改消费组名称。

“group.id”表示消费组的名称，您可以根据实际情况修改。

# consumer group id  
group.id=test-consumer-group
4. 按“Esc”，然后输入以下命令，按“Enter”，保存并退出“consumer.properties”文件。

:wq

消费消息示例如下：

```
[root@ecs-kafka bin]# ./kafka-console-consumer.sh --bootstrap-server 10.xx.xx.45:9095,10.xx.xx.127:9095,10.xx.xx.103:9095 --topic topic-demo --group order-test --from-beginning --consumer.config ../config/consumer.properties
Hello
DMS
Kafka!
^CProcessed a total of 3 messages
[root@ecs-kafka bin]#
```

如需停止消费使用Ctrl+C命令退出。

----结束

相关文档

- [Kafka连接问题排查和解决](#)
- [Kafka实例的内网连接地址可以修改吗？](#)
- [不同实例中，使用的SSL证书是否一样？](#)

5.6 使用控制台连接 Kafka

本章节介绍Kafka实例如何在控制台生产消息。此功能通过将指定消息发送给Kafka实例，验证业务逻辑是否正常。

前提条件

- 仅在实例的“状态”为“运行中”时，才可以在Topic中生产消息。

- 已[创建Topic](#)。

## 在控制台生产消息

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例名称，进入实例详情页面。
- 步骤4** 在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5** 通过以下任何一种方法，进入生产消息页面。
- 在待生产消息的Topic所在行，单击“更多 > 生产消息”（实例已开启SASL）/“生产消息”（实例未开启SASL），弹出“生产消息”对话框。
  - 单击待生产消息的Topic名称，进入Topic详情页面。单击右上角的“生产消息”，弹出“生产消息”对话框。
- 步骤6** 参考[表5-38](#)，设置消息的相关参数。

表 5-38 消息参数

| 参数名称    | 说明                                                                                                                                         |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------|
| 消息内容    | 输入消息的内容。长度为0~128000B。                                                                                                                      |
| 消息Key   | 输入消息的Key。                                                                                                                                  |
| 发送到指定分区 | 选择是否开启消息发送到指定的分区。 <ul style="list-style-type: none"><li>• 关闭：消息基于Key的hash值选择分区。</li><li>• 开启：消息将发送到指定的分区内。开启“发送到指定分区”后，需要选择分区ID。</li></ul> |

- 步骤7** 单击“确定”，完成消息的发送。
- 消息发送成功后，您可以在“实例管理 > 消息查询”页面，查看已成功发送的消息。
- 结束

## 相关文档

如果您希望通过API在控制台发送指定消息到Kafka实例中，请参考[Kafka生产消息](#)。

# 6 管理消息

## 6.1 查看 Kafka 消息

客户端连接Kafka实例生产消费过程中，如果遇到消息丢失或消息消费失败等问题，可以通过消息查询功能来查看特定消息的内容和相关属性，帮助排查问题根因。

您可以查看指定Topic不同分区的偏移量、消息大小、创建时间以及消息正文。

### 约束与限制

- 当Topic中的数据量比较大时，单副本Topic查询消息可能会报“内部服务错误”，建议根据数据量适当减小查询时间范围。
- 使用消息内容或KEY查询消息时，由于查询资源和性能限制，最大搜索范围为10000条消息且所有消息的总大小不超过200MB，最多返回满足查询要求的前10条消息。
- 在所有分区查询消息时，由于查询资源和性能限制，最多返回500条数据。在单分区查询消息时，没有此限制。
- 控制台只能展示4KB以内的消息，超过4KB的消息，请单击“下载消息”，下载到本地查看消息完整的内容。

### 查看 Kafka 消息

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 通过以下任意一种方法，进入查询消息页面。

- **查看任意Topic的消息：**在左侧导航栏选择“实例管理 > 消息查询”，进入消息列表页面。
- **查看指定Topic的消息：**在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。单击待查看消息的Topic名称，进入Topic详情页面。单击“消息查询”，进入“消息查询”页签。

**步骤5** Kafka实例支持两种查询消息的方法：按偏移量查询和按创建时间查询。

- 如果您知道消息所在Topic的分区和偏移量，可以使用按偏移量查询消息，具体查询参数参见表6-1。

表 6-1 按偏移量查询消息

| 参数名称    | 说明                                               |
|---------|--------------------------------------------------|
| Topic名称 | 仅从“实例管理 > 消息查询”路径进入时，才显示此参数。<br>选择待查询消息的Topic名称。 |
| 分区      | 输入消息所在的分区。                                       |
| 偏移量     | 输入消息的偏移量。                                        |

- 如果您知道消息发送时间段，但是不知道消息偏移量，可以使用按创建时间查询消息，具体查询参数参见表6-2。

表 6-2 按创建时间查询消息

| 参数名称    | 说明                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Topic名称 | 仅从“实例管理 > 消息查询”路径进入时，才显示此参数。<br>选择待查询消息的Topic名称。                                                                                                                                                                                                                                                                                                                                                                                         |
| 分区      | 输入消息所在的分区。<br>如果未设置具体分区，查询结果显示Topic所有分区的消息。<br>基于分区的消息查询限制说明： <ul style="list-style-type: none"><li><b>所有分区查询限制：</b>由于查询资源和性能限制，最多返回<b>500条</b>消息，且所有消息总大小不超过<b>200MB</b>。<br/><b>500条消息的获取过程：</b>后台按分区顺序（从分区0开始）依次拉取各分区最新的500条消息，若总大小超过200MB则立即终止查询。然后将已查询到的消息按时间排序，最终返回<b>时间最新的500条消息</b>。</li><li><b>单分区查询限制：</b>没有500条/200MB的限制，可以完整获取该分区内符合时间条件的<b>全部消息</b>。</li></ul> <b>大数据量查询建议：</b> 通过 <b>缩小查询时间范围</b> 或 <b>指定分区</b> 查询，提升查询效率。 |
| KEY查询   | 输入消息的KEY，查询结果为包含KEY的所有消息。<br>例如：Topic包含两条消息，KEY分别为“abc”和“abcd”，在“KEY查询”中输入“abc”，查询结果会同时返回这两条消息。<br>基于KEY的消息查询限制说明：由于查询资源和性能限制，最大搜索 <b>10000条</b> 消息且所有消息总大小不超过 <b>200MB</b> ，最多返回 <b>包含KEY的前10条消息</b> 。                                                                                                                                                                                                                                  |

| 参数名称 | 说明                                                                                                                                                                                                                                                                                                                                                  |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 内容查询 | 支持设置多个查询条件，查询结果为满足所有查询条件的消息。<br>设置查询条件的方法如下：<br>1. 在“查询内容”输入框中，单击鼠标左键，选择“包含”/“不包含”。<br>2. 输入消息正文的关键字，按Enter。<br>3. 如果需要设置多个查询条件，重复执行 <a href="#">步骤5.a</a> 和 <a href="#">步骤5.b</a> 。<br>基于内容的消息查询限制说明：由于查询资源和性能限制，最大搜索 <b>10000</b> 条消息且所有消息总大小不超过 <b>200MB</b> ，最多返回符合关键字的前 <b>10</b> 条消息。<br>查询超大消息（即单条消息超过20KB）或时间跨度大的消息时，建议将消息下载到本地，在本地进行关键字查询。 |
| 创建时间 | 消息的创建时间，设置待查询消息的时间范围。<br><b>警告</b><br>当Topic中的数据量比较大时，单副本Topic查询消息可能会报“内部服务错误”，建议根据数据量适当减小查询时间范围。                                                                                                                                                                                                                                                   |

**步骤6** 单击“搜索”，查询消息。

查询结果如下：

图 6-1 查询 Topic 消息

| Topic 名称         | 分区 | 偏移量 | 消息大小 (B) | 创建时间                          | 操作                     |
|------------------|----|-----|----------|-------------------------------|------------------------|
| topic-1081992957 | 0  | 1   | 8        | 2024/05/20 14:12:08 GMT+08:00 | <a href="#">查看消息正文</a> |
| topic-1081992957 | 0  | 0   | 9        | 2024/05/20 14:11:57 GMT+08:00 | <a href="#">查看消息正文</a> |

表 6-3 消息的参数说明

| 参数名称    | 说明                                                                   |
|---------|----------------------------------------------------------------------|
| Topic名称 | 消息所在的Topic名称。                                                        |
| 分区      | 消息所在的分区。                                                             |
| 偏移量     | 消息在分区中的位置。                                                           |
| 消息大小    | 消息存入磁盘的大小，单位为B。                                                      |
| 创建时间    | 消息的创建时间。创建时间由生产客户端在生产消息时通过CreateTime指定的，如果生产消息时没有设置此参数，创建时间会默认为1970。 |

**步骤7** 单击“查看消息正文”，弹出“查看消息正文”对话框，查看消息的内容，包括Topic名称、分区、偏移量、创建时间和消息正文。

控制台只能展示4KB以内的消息，超过4KB的消息，请单击“下载消息”，下载到本地查看消息完整的内容。

**步骤8** （可选）如果需要恢复默认设置，单击“重置”。

----结束

## 相关文档

- 如果您希望通过API查看Kafka消息，请参考[查询消息](#)。
- [为什么消息创建时间显示1970？](#)
- [为什么Console页面上，消息查询查不到消息？](#)

## 6.2 修改 Kafka 消息老化时间

老化时间即消息的最长保留时间，消费者必须在此时间结束前消费消息，否则消息将被删除。已删除的消息，无法被消费。

Topic老化时间默认为72小时，创建Topic成功之后，您可以根据业务需要修改Topic的老化时间。修改老化时间，不会影响业务。

修改老化时间可以通过以下方式实现：

- 在“实例管理 > Topic管理”中，修改老化时间。
- 在“实例管理 > 配置参数”中，修改“log.retention.hours”参数值，具体步骤请参考[修改Kafka实例配置参数](#)。

如果Topic已设置老化时间，则“实例管理 > 配置参数”中的log.retention.hours值对此Topic不生效。仅当Topic未设置老化时间时，该参数值才会生效。例如：Topic01设置的老化时间为60小时，log.retention.hours值为72小时，Topic01实际的老化时间仍为60小时。

## 约束与限制

- 单机实例只支持在“实例管理 > Topic管理”中，修改老化时间。
- 批量修改老化时间时，每次最多修改50个Topic的老化时间。

## 修改 Kafka 消息老化时间

Kafka控制台支持修改单个Topic的消息老化时间，也支持批量修改Topic的消息老化时间。

### 修改单个 Topic 消息老化时间

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

**步骤5** 在待修改老化时间的Topic所在行，单击“编辑”。

**步骤6** 在“编辑Topic”对话框中，输入老化时间（取值范围：1~720），单击“确定”。  
设置成功后，在Topic列表页面，查看修改后的老化时间。

----结束

批量修改 Topic 消息老化时间

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。
- 步骤4** 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。
- 步骤5** 勾选待修改老化时间的Topic，单击信息栏左上侧的“批量编辑Topic”，弹出“批量编辑Topic”对话框。
- 步骤6** 在“批量操作”区域，勾选“老化时间（小时）”，并输入老化时间（取值范围：1~720）。在“批量数据预览”区域，查看修改前和修改后的老化时间。确认无误后，单击“确定”。

图 6-2 批量修改老化时间

批量编辑Topic

勾选需要修改的配置将为所有选中的topic统一修改配置参数。

批量操作

☐ 分区数

☒ 老化时间（小时）

-

72

+

☐ 同步复制

☐ 同步落盘

☐ 消息时间戳类型

☐ 批处理消息最大值（字节）

批量数据预览

| Topic 名称    | 老化时间（小时） 修改后/前 |
|-------------|----------------|
| topic-doc01 | 72<br>81       |
| topic-doc02 | 72<br>73       |
| topic-doc03 | 无变化<br>72      |

总条数：3

10

<

1

>

设置成功后，在Topic列表页面，查看修改后的老化时间。

----结束

## 相关文档

- 如果您希望通过API修改消息老化时间，请参考[修改Kafka实例Topic](#)。
- [消息超过老化时间，消息仍存在的原因](#)

## 6.3 删除 Kafka 消息

本章节介绍如何在控制台删除Topic中存储的消息。

### 约束与限制

消息删除后无法恢复，请谨慎操作。

### 前提条件

删除消息前，请先在消费代码中设置“auto.offset.reset”参数。“auto.offset.reset”用来指定当Kafka中没有初始偏移量或者当前偏移量不存在（例如当前偏移量已被删除）时，消费者的消费策略。取值如下：

- latest：偏移量自动被重置到最晚偏移量。
- earliest：偏移量自动被重置到最早偏移量。
- none：向消费者抛出异常。

如果将此配置设置为latest，新增分区时，生产者可能会在消费者重置初始偏移量之前开始向新增加的分区发送消息，从而导致部分消息丢失。

### 删除 Kafka 消息

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

**步骤5** 通过以下任意一种方法，进入删除消息页面。

- 在待删除消息的Topic所在行，单击“更多 > 消息删除”，弹出“消息删除”对话框。
- 单击待删除消息的Topic名称，进入Topic详情页面。单击右上角的“消息删除”，弹出“消息删除”对话框。

**步骤6** 设置消息删除参数，如[表6-4](#)所示。

图 6-3 消息删除

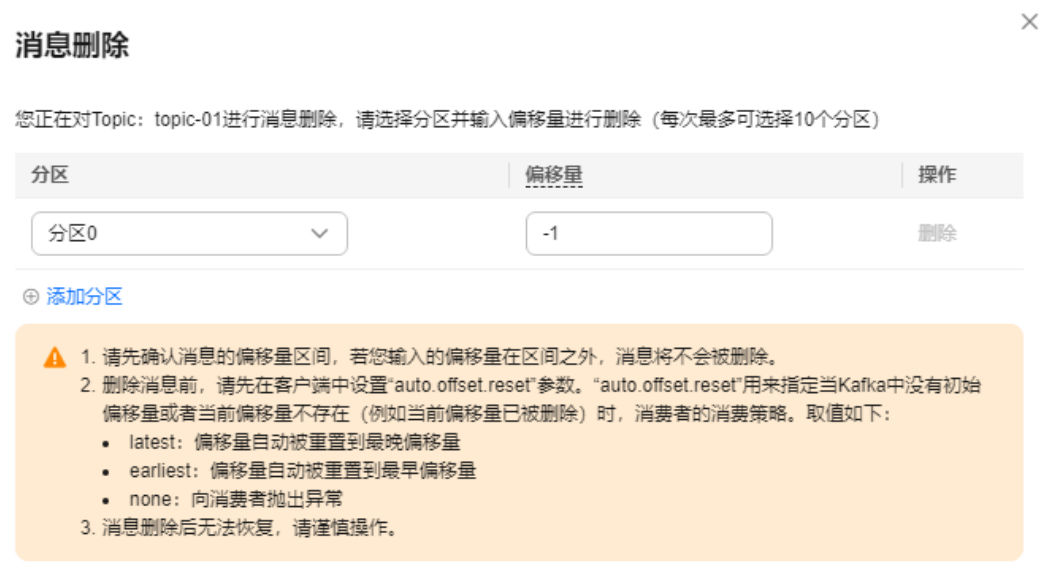


表 6-4 消息删除参数说明

| 参数  | 说明                                                                                                                                                                                                                  |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 分区  | 选择消息所在的分区编号。                                                                                                                                                                                                        |
| 偏移量 | 输入偏移量, 最早偏移量到输入偏移量之前的数据将被删除。<br>例如: 最早偏移量为2, 输入的偏移量为5, 此时会删除偏移量为2到4的消息。<br><b>警告</b> <ul style="list-style-type: none"><li>如果“偏移量”设置为“-1”, 表示删除分区中所有的消息。</li><li>如果您输入的偏移量不在指定分区的最早偏移量和最晚偏移量之间, 消息将不会被删除。</li></ul> |

如果需要删除多个分区的信息, 单击“添加分区”, 设置需要删除消息的分区和偏移量。每次最多可选择10个分区。

**步骤7** 单击“确定”, 弹出“清理结果”对话框, 单击“确定”, 完成消息的删除。

图 6-4 清理结果



**步骤8** 验证消息是否已成功删除。

在“实例管理 > 消息查询”页面使用偏移量查询，输入如下参数，如果查询不到消息，表示已成功删除消息。

- 分区：输入[步骤6](#)中的分区编号。
- 偏移量：如果[步骤6](#)中的偏移量设置为“-1”，随机输入一个偏移量。如果[步骤6](#)中的偏移量不为“-1”，输入小于此偏移量的整数。

----结束

## 相关文档

如果您希望通过API删除消息，请参考[Kafka删除消息](#)。

## 6.4 诊断 Kafka 消息积压

在消息处理过程中，如果客户端的消费速度跟不上服务端的发送速度，未处理的消息会越来越多，这部分消息就被称为堆积消息。消息没有被及时消费就会产生消息堆积，从而会造成消息消费延迟。

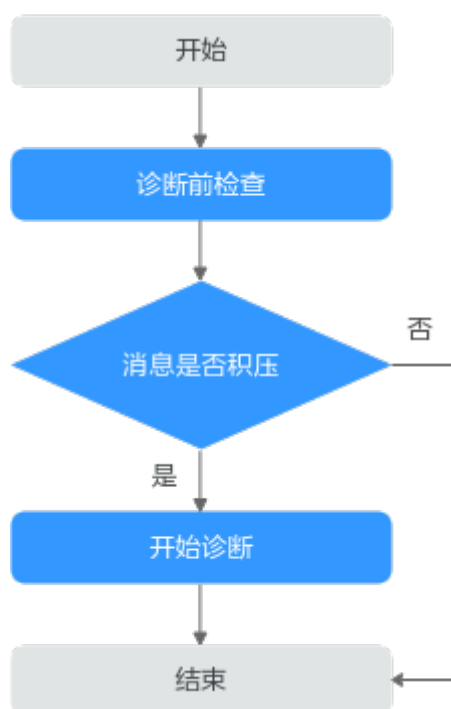
分布式消息服务Kafka版在控制台提供了消息积压诊断功能，如果消息存在堆积，诊断后会生成诊断记录，诊断记录显示消息堆积可能的原因、受影响的分区或者broker，以及解决消息堆积的建议。

## 前提条件

- 已创建Kafka实例，并且消费组正在以非assign方式消费消息中。
- 一个消费组正在诊断中时，无法对其他消费组进行诊断，也无法对此消费组消费的其他Topic进行诊断。

## 操作流程

图 6-5 消息积压诊断流程图



步骤一：诊断前检查

- 步骤1 登录Kafka控制台。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“分析与诊断 > 消息积压诊断”，进入消息积压诊断页面。
- 步骤5 选择需要诊断的消费组名称和Topic名称，单击“诊断前检查”。

检查成功后，在页面上方显示“预检查成功”，并显示内存使用率、CPU利用率、分区订阅关系、消息积压情况和流量突增情况的检查结果。

图 6-6 诊断前检查



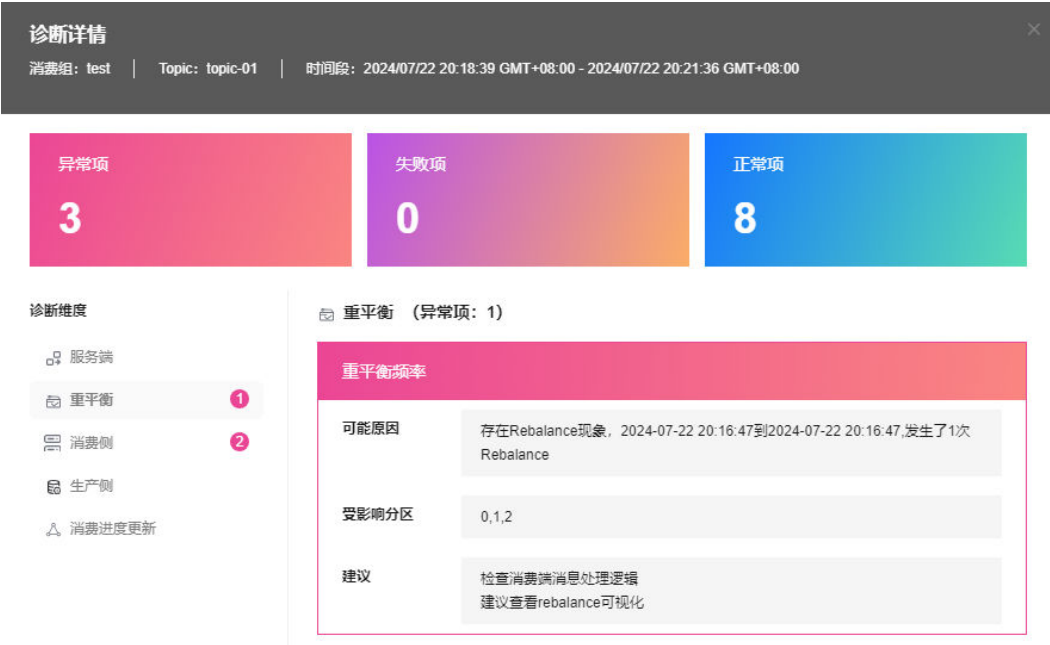
Topic所有分区的总消息堆积数小于2000时，“消息积压情况”显示无消息积压，此时无法进行消息积压诊断。“消息积压情况”显示有消息积压，并且消费组正在以非assign方式消费消息中，才能进行积压诊断。

----结束

步骤二：开始诊断

- 步骤1 单击“开始诊断”，在“诊断记录”区域，显示一条“状态”为“诊断中”的记录。当“状态”变为“诊断成功”时，表示消息积压诊断已经完成。
- 步骤2 在诊断记录所在行，单击“查看详情”，进入“诊断详情”页面。
- 步骤3 在页面顶部查看消息组异常项、失败项和正常项的数量。在“诊断维度”区域，单击包含异常项的维度名称，例如：重平衡，查看导致异常的可能原因、受影响的分区或者broker，以及解决消息堆积的建议。

图 6-7 诊断详情



----结束

相关文档

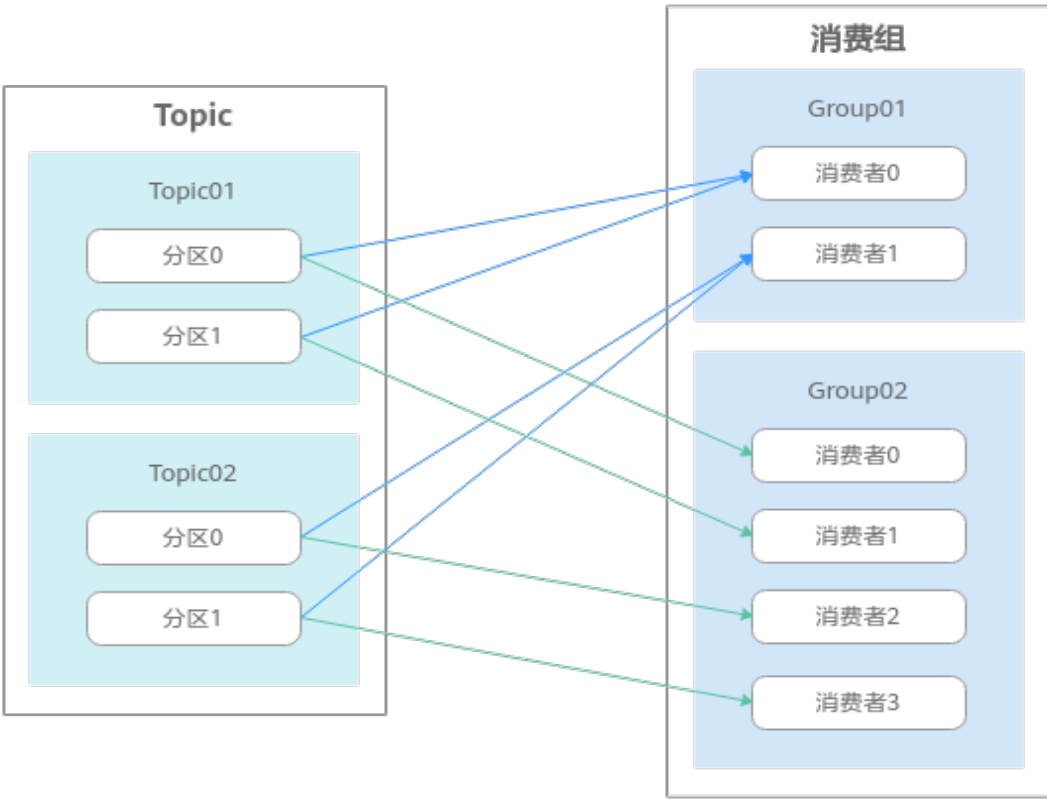
如果您希望通过API查看诊断报告详情, 请参考[查询诊断报告详情](#)。

# 7 管理消费组

## 7.1 创建 Kafka 消费组

消费者是从Topic订阅消息的一方，消费组是由一个或多个消费者组成的。同一个消费组内，一个消费者可同时消费多个分区，但一个分区在同一时刻只能被一个消费者消费。

图 7-1 消费关系示例图



“auto.create.groups.enable”表示在消费者进入未创建的消费组时，会自动创建此消费组。

- 如果在配置参数中设置“auto.create.groups.enable”为“false”，在消费消息前，需要先创建消费组，否则会导致消费失败。
- 如果在配置参数中设置“auto.create.groups.enable”为“true”，系统会自动创建消费组，无需手动创建。

本章节介绍如何在控制台创建消费组，此操作不会导致Kafka实例重启。

约束与限制

- 2023年4月25日及以后创建的实例，支持在控制台创建消费组。
- 如果“auto.create.groups.enable”设置为“true”，当消费组的状态为“EMPTY”且从未提交过offset，系统将在十分钟后自动删除该消费组。
- 如果“auto.create.groups.enable”设置为“false”，系统不会自动删除消费组。如果需要删除消费组，需要您手动删除。
- 如果消费组从未提交过offset，当Kafka实例重启后，该消费组会被删除。

创建 Kafka 消费组

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5 单击“创建消费组”，弹出“创建消费组”对话框。
- 步骤6 参考[表7-1](#)设置消费组参数，单击“确定”。

表 7-1 消费组参数说明

| 参数    | 说明                                                                                                        |
|-------|-----------------------------------------------------------------------------------------------------------|
| 消费组名称 | 自定义名称，但需要符合命名规则：只能由英文字母、数字、下划线开头，且只能英文字母、数字、句点、中划线、下划线组成，长度3~64个字符。<br>消费组名称开头包含特殊字符，例如#号“#”，会导致监控数据无法展示。 |
| 描述    | 消费组的描述信息，长度0~200个字符。                                                                                      |

在消费组列表页面，查看新创建的消费组。

----结束

相关文档

- 如果您希望通过API创建消费组，请参考[创建消费组](#)。
- [实例消费正常，为什么看不到消费者？](#)

## 7.2 查看 Kafka 消费组信息

消费组创建成功后，可查看消费组配置信息和状态信息。

### 查看消费组列表

您可以在Kafka控制台查看消费组列表，也可以通过客户端查看消费组列表。

#### 查看消费组列表（控制台）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 消费组列表页面显示消费组名称、消费组状态、Coordinator (ID)和描述。  
Coordinator (ID)显示的数字表示Coordinator组件所在的代理，消费组的状态如下。
- DEAD：消费组内没有任何成员，且没有任何元数据。
  - EMPTY：消费组内没有任何成员，存在元数据。
  - PREPARING\_REBALANCE：准备开启Rebalance。
  - COMPLETING\_REBALANCE：所有成员加入消费组。
  - STABLE：消费组内成员可以正常消费。

图 7-2 消费组列表

| <input type="checkbox"/> 消费组名称 | <input type="checkbox"/> 消费组状态 | <input type="checkbox"/> Coordinator (ID) | <input type="checkbox"/> 描述 | 操作                                    |
|--------------------------------|--------------------------------|-------------------------------------------|-----------------------------|---------------------------------------|
| <input type="checkbox"/> test  | EMPTY                          | 2                                         | --                          | <a href="#">编辑</a> <a href="#">删除</a> |

- 步骤5 （可选）如果需要搜索某个消费组，在搜索框中，输入消费组名称、消费组状态、Coordinator (ID)、堆积数、描述或者关键字，按“Enter”。
- 步骤6 （可选）如果需要刷新消费组列表，在右上角单击 。

----结束

#### 查看消费组列表（Kafka 客户端）

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令查询消费组列表。  

```
./kafka-consumer-groups.sh --bootstrap-server {connection-address} --list
```

  
connection-address表示Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。  
示例如下：  

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server 192.168.xx.xx:9092,192.xx.xx.212:9092,192.xx.xx.147:9092 --list
test
__consumer-group-dial-test
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤查询消费组列表。
  - a. （可选）如果已经设置了用户名和密码，以及SSL证书配置，请跳过此步骤，执行**b**。否则请执行以下操作。  
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加用户名和密码，以及SSL证书配置。
  - b. 在Kafka客户端的“/bin”目录下，通过以下命令查询消费组列表。  
`./kafka-consumer-groups.sh --bootstrap-server {connection-address} --list --command-config ../config/{ssl-user-config.properties}`

表 7-2 查询消费组列表参数说明

| 参数名称                       | 说明                                     |
|----------------------------|----------------------------------------|
| connection-address         | Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。 |
| ssl-user-config.properties | 配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。   |

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --list --command-config ../config/ssl-
user-config.properties
test
__consumer-group-dial-test
[root@ecs-kafka bin]#
```

相关文档

如果您希望通过API查看消费组列表，请参考[查询所有消费组](#)。

7.3 查看 Kafka 消费者信息

如果消费组中存在正在连接Kafka实例的消费者，可查看消费组中该消费者的连接信息。

约束与限制

- Kafka Manager由于缓存原因，显示的消费者连接地址可能不是当前消费者连接地址，建议重启Kafka Manager解决。
- 2023年5月17日及以后创建的实例不提供Kafka Manager功能，不支持在Kafka Manager中查看消费者连接地址。

前提条件

消费组中的消费者处于连接Kafka实例时（即消费组状态为“STABLE”），才可以查看消费者列表和消费者连接地址。

查看消费者列表

您可以在Kafka控制台查看消费者列表，也可以通过客户端查看消费者列表。

查看消费者列表（控制台）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5 单击待查看消费者列表的消费组名称，进入消费组详情页。
- 步骤6 在“消费者列表”页签，查看消费者列表。  
在消费者列表中，可以查看消费者ID、消费者地址和客户端ID。
- 步骤7 （可选）如果需要查询某个消费者，在搜索框中，输入消费者ID，按“Enter”。
- 结束

查看消费者列表（客户端）

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令查询消费者列表。  
`./kafka-consumer-groups.sh --bootstrap-server {connection-address} --group {group-name} --members --describe`

表 7-3 查看消费者列表参数说明

| 参数名称               | 说明                                     |
|--------------------|----------------------------------------|
| connection-address | Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。 |
| group-name         | 消费组名称。                                 |

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --group test --members --describe
GROUP CONSUMER-ID HOST CLIENT-ID
#PARTITIONS
test console-consumer-571a64fe-b0c4-47ce-833d-9e0da5a88d14 /192.168.0.215 console-
consumer 3
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤查询消费者列表。
  - （可选）如果已经设置了用户名和密码，以及SSL证书配置，请跳过此步骤，执行**b**。否则请执行以下操作。  
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加用户名和密码，以及SSL证书配置。
  - 在Kafka客户端的“/bin”目录下，通过以下命令查询消费者列表。  
`./kafka-consumer-groups.sh --bootstrap-server {connection-address} --group {group-name} --members --describe --command-config ../config/{ssl-user-config.properties}`

表 7-4 查看消费者列表参数说明

| 参数名称                       | 说明                                     |
|----------------------------|----------------------------------------|
| connection-address         | Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。 |
| group-name                 | 消费组名称。                                 |
| ssl-user-config.properties | 配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。   |

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --group test --members --describe --
command-config ../config/ssl-user-config.properties

GROUP CONSUMER-ID HOST CLIENT-ID
#PARTITIONS
test console-consumer-566d0c82-07d3-4d87-9a6e-f57a9bc9fc69 /192.168.0.215 console-
consumer 3
[root@ecs-kafka bin]#
```

查看消费者连接地址

您可以在Kafka控制台查看消费者连接地址，也可以通过Kafka Manager查看消费者连接地址。

查看消费者连接地址（控制台）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5 单击待查看消费者连接地址的消费组名称，进入消费组详情页。
- 步骤6 在“消费者列表”页签，查看消费者连接地址。

图 7-3 消费者列表

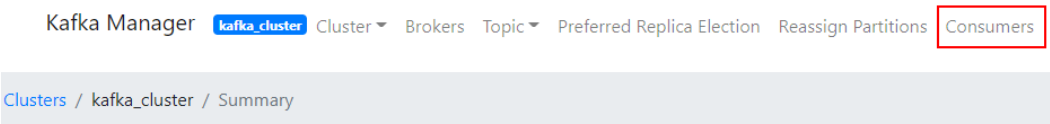


----结束

查看消费者连接地址（Kafka Manager）

- 步骤1 登录Kafka Manager。
- 步骤2 单击“kafka\_cluster”，进入集群详情页。
- 步骤3 在顶部导航栏单击“Consumers”，进入消费组列表页面。

图 7-4 导航栏



- 步骤4 单击待查看消费者连接地址的消费组名称，进入消费组订阅的Topic列表页面。

图 7-5 消费组列表页面

| Consumers       |      |                                                                         |  |
|-----------------|------|-------------------------------------------------------------------------|--|
| Show 10 entries |      |                                                                         |  |
| Consumer        | Type | Topics it consumes from                                                 |  |
| group01         | KF   | topic-01: (100% coverage, 180016 lag)<br>topic-02: (0% coverage, 0 lag) |  |
| group02         | KF   | topic-02: (0% coverage, 0 lag)                                          |  |
| group11         | KF   | topic-01: (100% coverage, 363016 lag)                                   |  |

- 步骤5 单击待查看消费者连接地址的Topic名称，进入Topic详情页。

图 7-6 消费组订阅的 Topic 列表页面

| Consumed Topic Information |                      |
|----------------------------|----------------------|
| Topic                      | Partitions Covered % |
| topic-01                   | 100                  |
| topic-02                   | 0                    |

- 步骤6 在“Consumer Instance Owner”中，查看消费者连接地址。

图 7-7 Topic 详情页

| Partition | LogSize | Consumer Offset | Lag    | Consumer Instance Owner                                        |
|-----------|---------|-----------------|--------|----------------------------------------------------------------|
| 0         | 33,333  | 0               | 33,333 | consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1/10.204.177.102 |
| 1         | 33,334  | 0               | 33,334 | consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1/10.214.177.102 |
| 2         | 33,333  | 0               | 33,333 | consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1/10.234.177.102 |

----结束

相关文档

如果您希望通过API查看消费者列表和连接地址，请参考[查询指定消费组](#)。

7.4 查看和重置 Kafka 消费进度

消费进度表示消费者的消费位置，本章节介绍如何查看和重置消费进度，重置消费进度即修改消费者的消费位置。

约束与限制

重置消费进度可能会导致重复消费，请谨慎操作。

前提条件

Kafka实例不支持在线重置消费进度，请先将待重置消费进度的消费组停止消费，然后重置消费进度。停止待重置消费进度的消费者客户端后，需要等待一段时间（即ConsumerConfig.SESSION\_TIMEOUT\_MS\_CONFIG配置的时间，默认为1000毫秒），服务端才会认为此消费者已下线。

查看消费进度

您可以在Kafka控制台查看消费进度，也可以通过客户端查看消费进度。

查看消费进度（控制台）

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5 单击待查看消费进度的消费组名称，进入消费组详情页。
- 步骤6 在“消费进度”页签，查看此消费组订阅的Topic列表、Topic的分区数、Topic中消息的总堆积数、Topic每个分区的消费进度。

图 7-8 消费进度



表 7-5 消费进度参数说明

| 参数名称    | 说明                                                                                                                                |
|---------|-----------------------------------------------------------------------------------------------------------------------------------|
| Topic名称 | 消费组订阅的Topic名称。                                                                                                                    |
| 分区数     | Topic的分区数。                                                                                                                        |
| 总堆积数    | Topic中未被消费组消费的消息数。<br>总堆积数为采集时刻的瞬时值，而监控指标中的堆积数采集周期为1分钟，由于采集周期不同，可能会导致数值存在差异。具体现象请参见 <a href="#">消费进度显示有堆积消息，为什么监控中显示消息堆积数为0</a> 。 |
| 分区编号    | Topic的分区号。                                                                                                                        |
| 堆积数     | 该分区中未被消费组消费的消息数。                                                                                                                  |
| 偏移量     | 该分区的偏移量。                                                                                                                          |
| 最大消息位置  | 该分区的最大消息位置。                                                                                                                       |
| 消费者ID   | 消费该分区消息的消费者ID。                                                                                                                    |
| 消费者地址   | 消费该分区消息的消费者地址。                                                                                                                    |
| 客户端ID   | 客户端标识符，该客户端用于连接Kafka实例并消费该分区的消息。                                                                                                  |

- 步骤7（可选）如果需要查询某个Topic的消费进度，在搜索框中，输入Topic名称，按“Enter”。
- 步骤8（可选）如果需要导出消费进度到本地，参考以下任意一种方法操作。
- 勾选待导出的Topic，单击“导出 > 导出已选中数据到XLSX”，导出指定Topic的消费进度。
  - 单击“导出 > 导出全部数据到XLSX”，导出全部Topic的消费进度。

----结束

查看消费进度（Kafka 客户端）

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令查询消费进度。

connection-address表示Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --offsets --describe --all-groups
```

Consumer group '\_consumer-group-dial-test' has no active members.

| GROUP                      | TOPIC           | PARTITION | CURRENT-OFFSET | LOG-END-OFFSET | LAG |
|----------------------------|-----------------|-----------|----------------|----------------|-----|
| CONSUMER-ID                | HOST            | CLIENT-ID |                |                |     |
| __consumer-group-dial-test | __dms_dial_test | 0         | 350            | 350            | 0   |
| -                          | -               | -         | -              | -              | -   |
| __consumer-group-dial-test | __dms_dial_test | 1         | 350            | 350            | 0   |
| -                          | -               | -         | -              | -              | -   |
| __consumer-group-dial-test | __dms_dial_test | 2         | 350            | 350            | 0   |
| -                          | -               | -         | -              | -              | -   |

Consumer group 'test' has no active members.

| GROUP       | TOPIC    | PARTITION | CURRENT-OFFSET | LOG-END-OFFSET | LAG |   |   |   |
|-------------|----------|-----------|----------------|----------------|-----|---|---|---|
| CONSUMER-ID | HOST     | CLIENT-ID |                |                |     |   |   |   |
| test        | topic-01 | 0         | 5              | 5              | 0   | - | - | - |
| test        | topic-01 | 1         | 3              | 3              | 0   | - | - | - |
| test        | topic-01 | 2         | 10             | 10             | 0   | - | - | - |

- 已开启密文接入的Kafka实例，通过以下步骤查询消费进度。
  - a. （可选）如果已经设置了用户名和密码，以及SSL证书配置，请跳过此步骤，执行**b**。否则请执行以下操作。

在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加用户名和密码，以及SSL证书配置。
  - b. 在Kafka客户端的“/bin”目录下，通过以下命令查询消费进度。

```
./kafka-consumer-groups.sh --bootstrap-server {connection-address} --offsets --describe --all-groups --command-config ../config/{ssl-user-config.properties}
```

### 表 7-6 查看消费进度参数说明

| 参数名称                       | 说明                                     |
|----------------------------|----------------------------------------|
| connection-address         | Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。 |
| ssl-user-config.properties | 配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。   |

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --offsets --describe --all-groups --
command-config ../config/ssl-user-config.properties
```

Consumer group '\_consumer-group-dial-test' has no active members.

| GROUP                      | TOPIC           | PARTITION | CURRENT-OFFSET | LOG-END-OFFSET |   |
|----------------------------|-----------------|-----------|----------------|----------------|---|
| LAG                        | CONSUMER-ID     | HOST      | CLIENT-ID      |                |   |
| __consumer-group-dial-test | __dms_dial_test | 0         | 347            | 347            | 0 |
| -                          | -               |           |                |                |   |
| __consumer-group-dial-test | __dms_dial_test | 1         | 347            | 347            | 0 |

```
_consumer-group-dial-test __dms_dial_test 2 347 347 0
- - -

Consumer group 'test' has no active members.

GROUP TOPIC PARTITION CURRENT-OFFSET LOG-END-OFFSET LAG
CONSUMER-ID HOST CLIENT-ID
test topic-01 0 5 0 -
test topic-01 1 3 0 -
test topic-01 2 10 0 -
[root@ecs-kafka bin]#
```

重置消费进度

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5 单击待重置消费进度的消费组名称，进入消费组详情页。
- 步骤6 在“消费进度”页签，通过以下方法，重置消费进度。
  - 重置单个Topic所有分区的消费进度：在待重置消费进度的Topic后，单击“重置消费进度”。
  - 重置单个Topic中单个分区的消费进度：在待重置消费进度的Topic分区后，单击“重置消费进度”。
  - 重置所有Topic所有分区的消费进度：单击“重置所有消费进度”。
- 步骤7 在弹出的“重置消费进度”对话框中，参考[表7-7](#)，设置重置消费进度参数。

表 7-7 重置消费进度参数说明

| 参数   | 说明                                                                                                                                                    |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| 重置类型 | 选择重置类型： <ul style="list-style-type: none"><li>时间：重置消费进度到指定的时间。</li><li>偏移量：重置消费进度到指定的偏移量。</li></ul> “重置所有消费进度”只支持重置消费进度到指定时间。                         |
| 时间   | 当“重置类型”为“时间”时，需要设置此参数。<br>选择重置消费进度的时间点，重置完成后，将从此时间点开始消费。 <ul style="list-style-type: none"><li>最早：最早偏移量</li><li>自定义：自定义时间点</li><li>最晚：最晚偏移量</li></ul> |
| 偏移量  | 当“重置类型”为“偏移量”时，需要设置此参数。<br>选择重置消费进度的偏移量，此偏移量不能小于0，重置完成后，将从此偏移量开始消费。                                                                                   |

**步骤8** 单击“确定”，弹出确认对话框。

**步骤9** 单击“是”，完成消费进度的重置。

在“消费进度”页签，在已重置消费进度的Topic前单击，在“偏移量”列查看重置后的值。

----结束

## 相关文档

如果您希望通过API重置消费进度，请参考[重置消费组消费进度到指定位置](#)。

## 7.5 查看 Kafka 重平衡日志

Rebalance是重新分配消费组中的消费者和Topic分区之间订阅关系的过程。在Rebalance过程中消费组下的所有消费者都会停止消费，等待Rebalance完成。

可能会触发Rebalance的原因：

- 消费组成员个数发生变化，例如有新的消费者加入，或者有消费者退出。
- 消费组订阅的Topic个数发生变化。
- 消费组订阅的Topic分区数发生变化。

重平衡日志记录Rebalance的详情，包括Rebalance时间、原因和触发Rebalance的客户端等。本章节介绍如何在控制台查看重平衡日志。

重平衡日志存储在云日志服务（Log Tank Service，简称LTS）中，由LTS提供查询功能。

### 约束与限制

- 2023年4月6日前创建的实例，不支持重平衡日志功能。
- 单机实例不支持查看重平衡日志。
- 重平衡日志默认存储时间为七天，如果需要延长保留天数，可以在[LTS控制台修改日志组的存储时间](#)。
- 重平衡日志开启后会在LTS控制台创建对应的日志组、日志流和仪表盘。使用期间按照日志量收费，收费标准请参照[LTS价格详情](#)。
- 不同Kafka实例使用相同的日志组和日志流保存重平衡日志时，在每个Kafka实例中都能看到使用同一个日志组和日志流的所有Kafka实例的重平衡日志。

### 前提条件

- 开启重平衡日志需要您的账号拥有LTS服务的创建日志组和日志流的权限。
- Kafka实例处于“运行中”时，才能开启/关闭重平衡日志。

### 开启重平衡日志

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“分析与诊断 > 重平衡日志”，进入“重平衡日志”页面。
- 步骤5 单击“开启日志”，弹出“开启日志”对话框。
- 步骤6 单击“确定”，弹出“日志配置”对话框。
- 步骤7 根据实际情况决定是否开启日志配置，单击“确定”，自动跳转到“后台任务管理”页面，“状态”为“成功”时，表示开启重平衡日志成功。
  - 不开启：LTS后台会自动创建一个日志组以及对应的日志流。
  - 开启：选择存储日志文件“coordinator.log”的日志组和日志流。您还可以单击右侧的“查看日志组”，跳转到LTS控制台，查看或创建日志组和日志流。

----结束
- 查看重平衡日志
- 步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏单击“分析与诊断 > 重平衡日志”，进入“重平衡日志”页面。

步骤5 在“仪表盘”页签，查看消费组重平衡次数和原因。在“日志查询”页签，查看重平衡日志。

如果您需要搜索日志，请参考[进入搜索LTS日志页面](#)进行操作。

重平衡日志示例如下：
- ```
{
  "level": "INFO",
  "timestamp": "2023-03-23 17:23:22,906",
  "message": {
    "leaderId": "consumer-1-177817b6-1f29-4717-8a83-dda8eaab1635",
    "generationId": "1",
    "reason": "Assignment received from leader for group KMOffsetCache-dms-vm-fa3cf9d6-manager-shared-server-0 for generation 1",
    "groupId": "KMOffsetCache-dms-vm-fa3cf9d6-manager-shared-server-0",
    "coordinatorId": "0",
    "type": "END_REBALANCE",
    "group": "GroupMetadata(groupId=KMOffsetCache-dms-vm-fa3cf9d6-manager-shared-server-0, generation=1, protocolType=Some(consumer), currentState=CompletingRebalance, members=Map(consumer-1-177817b6-1f29-4717-8a83-dda8eaab1635 -> MemberMetadata(memberId=consumer-1-177817b6-1f29-4717-8a83-dda8eaab1635, clientId=consumer-1, clientHost=/172.31.2.168, sessionTimeoutMs=10000, rebalanceTimeoutMs=300000, supportedProtocols=List(range), )))"
  }
}
```
- 参数说明如[表7-8](#)所示。
- 表 7-8 重平衡参数说明
- | 参数名称 | 说明 |
|-------|------------------------|
| level | 重平衡日志的等级，只有“INFO”一种等级。 |
- 文档版本 01 (2025-09-17)

版权所有 © 华为云计算技术有限公司

154

参数名称	说明
timestamp	Rebalance时间。
leaderId	消费者Leader ID。
generationId	消费组Generation ID。Generation等同于消费组执行Rebalance的次数，每次Rebalance完成后，Generation都会增加1。
reason	触发Rebalance的原因。
groupId	消费组ID。
coordinatorId	Coordinator组件所在的Broker。
type	触发Rebalance的操作，取值如下： • JOIN_GROUP：新消费者加入消费组。 • OVER_CAPACITY：超过组限制。 • UPDATE_MEMBER：消费者更新元数据。 • PROTOCOL_CHANGE：协议变化。 • HEARTBEAT_EXPIRED：消费者心跳超时。 • SYNC_GROUP：同步分配计划。 • END_REBALANCE：重平衡结束。 • LEAVE_GROUP：消费者主动离开消费组。 • DELETE_GROUP：用户删除消费组。
group	消费组中消费者的信息。

----结束

关闭重平衡日志

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“分析与诊断 > 重平衡日志”，进入“重平衡日志”页面。
- 步骤5 在页面右上角单击“关闭日志”，弹出确认关闭对话框。
- 步骤6 单击“确定”，自动跳转到“后台任务管理”页面，当重平衡日志任务的“状态”为“成功”时，表示关闭重平衡日志成功。

关闭重平衡日志只是停止重平衡日志上报功能，LTS控制台的日志组和日志流仍然保留，还会继续收费，如不需要保留该日志，可以在LTS控制台删除对应的[日志组](#)和[日志流](#)。

----结束

相关文档

[为什么Kafka实例常常发生Rebalance（重平衡），导致消息拉取失败？](#)

7.6 修改 Kafka 消费组信息

消费组创建成功后，您可以根据业务需要修改消费组的描述信息。

修改 Kafka 消费组信息

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。

步骤5 通过以下任意一种方法，进入编辑消费组页面。

- 在待修改信息的消费组所在行，单击“编辑”，弹出“编辑”对话框。
- 单击待修改信息的消费组名称，进入消费组详情页面。单击右上角的“编辑”，弹出“编辑”对话框。

步骤6 修改描述信息，单击“确定”。

修改完成后，在消费组列表页面的“描述”列查看修改后的描述信息。

----结束

相关文档

如果您希望通过API修改消费组信息，请参考[修改指定消费组](#)。

7.7 配置 Kafka 自动创建消费组

开启“Kafka自动创建消费组”功能后，当消费者进入未创建的消费组时，系统会自动创建该消费组。若关闭此参数，则需在消费消息前手动创建消费组，否则会导致消费失败。

约束与限制

- 新创建的Kafka实例默认开启自动创建消费组功能。
- 修改Kafka自动创建消费组功能，不会重启Kafka实例。

配置 Kafka 自动创建消费组

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 通过以下任意一种方法，修改Kafka自动创建消费组功能。

- 在“概览”页面的“实例信息”区域，开启/关闭Kafka自动创建消费组功能。
- 在左侧导航栏单击“实例管理 > 配置参数”，进入配置参数页面。单击“auto.create.groups.enable”后的“编辑”，在“参数运行值”中开启/关闭Kafka自动创建消费组功能。

----结束

7.8 导出 Kafka 消费组列表

消费组创建成功后，可导出Kafka实例中消费组的列表信息。

导出 Kafka 消费组列表

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。

步骤5 通过以下任意一种方法，导出消费组列表。

- 勾选待导出的消费组，单击“导出 > 导出已选中数据到XLSX”，导出指定消费组列表。
- 单击“导出 > 导出全部数据到XLSX”，导出全部消费组列表。

----结束

7.9 删除 Kafka 消费组

Kafka实例支持通过以下两种方式删除消费组，您可以根据实际情况选择任意一种方式。

- 在控制台删除消费组
- 在[Kafka客户端](#)使用命令行工具删除消费组（确保Kafka实例版本与命令行工具版本相同）

约束与限制

- 若“auto.create.groups.enable”设置为“true”，当消费组的状态为“EMPTY”且从未提交过offset，系统将在十分钟后自动删除该消费组。
- 若“auto.create.groups.enable”设置为“false”，系统不会自动删除消费组。如果需要删除消费组，需要您手动删除。
- 若消费组从未提交过offset，当Kafka实例重启后，该消费组会被删除。
- 删除消费组会导致消费进度丢失，造成重新消费或重复消费等情况。

前提条件

待删除消费组的状态为“EMPTY”。

删除消费组

您可以在Kafka控制台删除消费组，也可以通过客户端删除消费组。

删除消费组（控制台）

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。
- 步骤4** 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5** 通过以下任意一种方法，删除消费组。
- 勾选消费组名称左侧的方框，可选一个或多个，单击信息栏左上侧的“删除消费组”。
 - 在待删除消费组所在行，单击“删除”。
 - 单击待删除消费组名称，进入消费组详情页。在页面右上角单击“删除”。
- 步骤6** 弹出“删除消费组”对话框，单击“确定”，完成消费组的删除。
- 已删除的消费组不再显示在消费组列表中，表明消费组已成功删除。
- 结束

删除消费组（Kafka 客户端）

以下操作命令以Linux系统为例进行说明：

- 未开启密文接入的Kafka实例，在Kafka客户端的“/bin”目录下，通过以下命令删除消费组。
`./kafka-consumer-groups.sh --bootstrap-server {connection-address} --delete --group {group-name}`

表 7-9 删除消费组参数说明

参数名称	说明
connection-address	Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。
group-name	消费组名称。

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9092,192.168.xx.xx:9092,192.168.xx.xx:9092 --delete --group group-01
Deletion of requested consumer groups ('group-01') was successful.
[root@ecs-kafka bin]#
```

- 已开启密文接入的Kafka实例，通过以下步骤删除消费组。
 - （可选）如果已经设置了SSL证书配置，请跳过此步骤，执行**步骤b**。否则请执行以下操作。
在Kafka客户端的“/config”目录中创建“ssl-user-config.properties”文件，参考**步骤3**增加SSL证书配置。

- b. 在Kafka客户端的“/bin”目录下，通过以下命令删除消费组。
- ```
./kafka-consumer-groups.sh --bootstrap-server {connection-address} --delete --group {group-name} --command-config ../config/{ssl-user-config.properties}
```

表 7-10 删除消费组参数说明

| 参数名称                       | 说明                                     |
|----------------------------|----------------------------------------|
| connection-address         | Kafka实例的连接地址，在Kafka控制台的“概览 > 连接信息”中获取。 |
| group-name                 | 消费组名称。                                 |
| ssl-user-config.properties | 配置文件的名称，此配置文件用于存放用户名和密码，以及SSL证书配置信息。   |

示例如下：

```
[root@ecs-kafka bin]# ./kafka-consumer-groups.sh --bootstrap-server
192.168.xx.xx:9093,192.168.xx.xx:9093,192.168.xx.xx:9093 --delete --group group-02 --command-
config ../config/ssl-user-config.properties
Deletion of requested consumer groups ('group-02') was successful.
[root@ecs-kafka bin]#
```

## 相关文档

如果您希望通过API删除消费组，请参考[删除指定消费组](#)。

## 7.10 取消 Kafka 消费组和 Topic 的订阅关系

消费者订阅Topic消费消息后，在消费组详情页可以查看消费进度。如果消费者不再消费某个Topic，您可以取消消费组与此Topic的订阅关系，避免由于此Topic的消息堆积，导致的告警误报。

### 约束与限制

- 已创建的1.1.0/2.3.0版本的Kafka实例可能不支持取消订阅功能，您可以通过[升级实例的内核版本](#)来解决此问题。
- 取消消费组和Topic的订阅关系会删除消费组对该Topic的消费位点信息，删除后无法恢复，请谨慎操作。

### 取消 Kafka 消费组和 Topic 的订阅关系

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。
- 步骤4** 在左侧导航栏选择“实例管理 > 消费组管理”，进入消费组列表页面。
- 步骤5** 单击待取消订阅关系的消费组名称，进入消费组详情页。

**步骤6** 在“消费进度”页签中，勾选待取消订阅关系的Topic，每次最多勾选**50个**，单击信息栏左上侧的“取消订阅”。

**步骤7** 在“取消订阅”对话框中，单击“是”。

取消订阅成功后，在“消费进度”页签，不再显示取消订阅的Topic。

----结束

# 8 配置流控

## 8.1 配置 Kafka 流控

Kafka支持对用户/客户端/Topic配置流量控制，控制生产/消费消息的上限速率。

用户/客户端的流控作用范围是整个Broker，Topic的流控作用范围是指定Topic。

### 约束与限制

- 2022年11月10日及以后创建的实例支持此功能。
- 单机实例不支持配置流控。

### 操作影响

- 当流控值达到上限后，会导致生产/消费的时延增大。
- 设置的流控值较小且生产者速率较大时，可能会造成生产超时、消息丢失，导致部分消息生产失败。
- 初始生产/消费的流量较大，如果设置一个较小的流控值，会导致生产/消费的时延增大、部分消息生产失败。建议逐次减半设置流控值，待生产/消费稳定后继续减半设置，直到设置为目标流控值。例如初始生产流量100MB/s，可先设置生产流控为50MB/s，待稳定后再修改为25MB/s，直到目标流控值。

### 前提条件

- 如果需要对用户进行流量控制，请在Kafka详情页[开启密文接入](#)功能。然后在控制台的“实例管理 > 用户管理”页面，获取用户名。
- 如果需要对指定客户端进行流量控制，请在客户端配置中获取client ID。
- 如果需要对指定Topic进行流量控制，请在控制台的“实例管理 > Topic管理”页面，获取Topic名称。

### 创建流控

以下介绍创建用户/客户端/Topic流控的步骤。

创建用户/客户端流控

- 步骤1 登录Kafka控制台。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 在左侧导航栏单击“实例管理 > 流控管理”，进入流控列表页面。
- 步骤5 在页面顶端单击“用户/客户端”，进入“用户/客户端”页签。
- 步骤6 在页面左上角单击“创建流控”，弹出“创建流控”对话框。
- 步骤7 设置流控参数。

图 8-1 创建用户/客户端流控

创建流控

 设置流控可能增大请求时延，造成生产超时、消息丢失等异常情况 [查看详情](#) 

温馨提示：用户名和客户端 ID 不可同时为空。

用户名

选择默认

选择默认表示对全部用户限流

客户端 ID

选择默认

选择默认表示对全部客户端限流

温馨提示：为空表示不限速率，但生产上限速率和消费上限速率不可同时为空。

生产上限速率

-

2

+

(MB/s)

消费上限速率

-

+

(MB/s)

表 8-1 流控参数说明

| 参数名称 | 说明                                                                          |
|------|-----------------------------------------------------------------------------|
| 用户名  | 输入前提条件中获取的用户名，对此用户进行流控。如果需要对所有用户进行流控，在“用户名”后，单击“选择默认”。<br>流控创建完后，无法修改“用户名”。 |

文档版本 01 (2025-09-17)

版权所有 © 华为云计算技术有限公司

162

| 参数名称   | 说明                                                                                  |
|--------|-------------------------------------------------------------------------------------|
| 客户端ID  | 输入前提条件中获取的客户端ID，对此客户端进行流控。如果需要对所有客户端进行流控，在“客户端ID”后，单击“选择默认”。<br>流控创建完后，无法修改“客户端ID”。 |
| 生产上限速率 | 设置生产上限速率，单位为MB/s。为空时，表示不设置速率。                                                       |
| 消费上限速率 | 设置消费上限速率，单位为MB/s。为空时，表示不设置速率。                                                       |

⚠ 注意

- 未开启密文接入的实例，在“创建流控”对话框中，不显示“用户名”。
- “用户名”和“客户端ID”不可同时为空。
- “生产上限速率”和“消费上限速率”不可同时为空。

- 步骤8** 单击“确定”，跳转到“后台任务管理”页面，当流控任务的“状态”为“成功”时，表示流控创建成功。
- 进入“实例管理 > 流控管理”页面，在“用户/客户端”页签中，通过以下方法查看新创建的流控。
- 未开启密文接入的实例：在搜索框中输入新创建的流控名称，按“Enter”。
  - 已开启密文接入的实例：单击页面左上角的“仅设置了用户名” / “仅设置了客户端ID” / “设置了用户名和客户端ID”，选择新创建的流控所属分类，然后在搜索框中输入新创建的流控名称，按“Enter”

图 8-2 查看新创建的流控



----结束

创建 Topic 流控

- 步骤1** 登录Kafka控制台。
- 步骤2** 在管理控制台左上角单击📍，选择Kafka实例所在的区域。
- 步骤3** 单击Kafka实例的名称，进入实例详情页面。

- 步骤4 在左侧导航栏单击“实例管理 > 流控管理”，进入流控列表页面。
- 步骤5 在页面顶端单击“Topic”，进入“Topic”页签。
- 步骤6 在页面左上角单击“创建流控”，弹出“创建流控”对话框。
- 步骤7 设置流控参数。

图 8-3 创建 Topic 流控

创建流控

⚠ 设置流控可能增大请求时延，造成生产超时、消息丢失等异常情况 [查看详情](#)

★ Topic 名称

topic01

温馨提示：为空表示不限速率，但生产上限速率和消费上限速率不可同时为空。

生产上限速率

—

2

+

(MB/s)

消费上限速率

—

+

(MB/s)

表 8-2 流控参数说明

| 参数名称    | 说明                                                |
|---------|---------------------------------------------------|
| Topic名称 | 输入指定Topic名称，对此Topic进行流控。<br>流控创建完后，无法修改“Topic名称”。 |
| 生产上限速率  | 设置生产上限速率，单位为MB/s。为空时，表示不设置速率。                     |
| 消费上限速率  | 设置消费上限速率，单位为MB/s。为空时，表示不设置速率。                     |

⚠ 注意

“生产上限速率”和“消费上限速率”不可同时为空。

- 步骤8 单击“确定”，跳转到“后台任务管理”页面，当流控任务的“状态”为“成功”时，表示流控创建成功。
- 进入“实例管理 > 流控管理”页面，在“Topic”页签的搜索框中输入新创建的流控名称，单击“Enter”，查看新创建的流控。
- 结束

## 编辑流控

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“实例管理 > 流控管理”，进入流控列表页面。

**步骤5** 在待修改的流控所在行，单击“编辑”，弹出“编辑流控”对话框。

**步骤6** 修改生产上限速率或者消费上限速率，单击“确定”，跳转到“后台任务管理”页面，当流控任务的“状态”为“成功”时，表示流控修改成功。

进入“实例管理 > 流控管理”页面，查看修改后流控的生产上限速率或者消费上限速率。

### 说明

“生产上限速率”和“消费上限速率”不可同时为空。

----结束

## 导出流控列表

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“实例管理 > 流控管理”，进入流控列表页面。

**步骤5** 导出流控列表。

- 指定用户/客户端流控：在“用户/客户端”页签，勾选待导出的用户/客户端流控，单击“导出 > 导出已选中数据到XLSX”。
- 全部用户/客户端流控：在“用户/客户端”页签，单击“导出 > 导出全部数据到XLSX”。
- 指定Topic流控：在“Topic”页签，勾选待导出的Topic流控，单击“导出 > 导出已选中数据到XLSX”。
- 全部Topic流控：在“Topic”页签，单击“导出 > 导出全部数据到XLSX”。

----结束

## 删除流控

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“实例管理 > 流控管理”，进入流控列表页面。

**步骤5** 在待删除的流控所在行，单击“删除”，弹出“删除流控”对话框。

**步骤6** 单击“确定”，跳转到“后台任务管理”页面，当流控任务的“状态”为“成功”时，表示成功删除流控。

----结束

## 相关文档

- 如果您希望通过API创建用户/客户端流控，请参考[创建用户/客户端流控配置](#)。
- 如果您希望通过API创建Topic流控，请参考[创建Topic流控配置](#)。

## 8.2 查看 Kafka 流控统计

如果Kafka配置了流量控制，可在控制台查看当前流控策略下，每个代理上用户/客户端/Topic的带宽使用量。

### 约束与限制

单机实例不支持查看流控监控。

### 查看带宽使用量

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“监控 > 流控监控”，进入流控监控页面。

**步骤5** 设置查询带宽使用量的参数。

图 8-4 查询带宽使用量的参数



表 8-3 查询带宽使用量的参数说明

| 参数   | 说明                                                                                                                                                                                                      |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 统计方式 | 设置带宽使用量的统计方式。 <ul style="list-style-type: none"><li>前n个：统计带宽使用量排名前x个的用户/客户端/Topic，x由您自行输入。</li><li>带宽速率：统计带宽速率大于x MB/s的用户/客户端/Topic，x由您自行输入。</li><li>带宽使用量：统计带宽使用量超过x %的用户/客户端/Topic，x由您自行输入。</li></ul> |
| 统计类型 | 设置带宽使用量的统计类型。 <ul style="list-style-type: none"><li>生产：统计生产消息的带宽使用量。</li><li>消费：统计消费消息的带宽使用量。</li></ul>                                                                                                 |
| 统计维度 | 设置带宽使用量的统计维度。 <ul style="list-style-type: none"><li>用户/客户端：统计用户/客户端的带宽使用量。</li><li>Topic：统计Topic的带宽使用量。</li></ul>                                                                                       |

**步骤6** 单击“查询”，查看每个代理上用户/客户端/Topic的带宽使用量。

----结束

# 9 管理实例

## 9.1 查看和修改 Kafka 实例基本信息

创建Kafka实例成功后，您可以在控制台查看Kafka实例的详细信息，并根据自己的业务情况对Kafka实例的部分参数进行调整，包括实例名称、描述、安全组和容量阈值策略等。

### 约束与限制

单机实例不支持修改Smart Connect和内网访问。

### 前提条件

Kafka实例的状态为“运行中”时，才能修改实例的基本信息。

### 查看 Kafka 实例的详细信息

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** Kafka实例支持通过筛选功能查询目标实例。当前支持的筛选条件为名称、状态、版本、实例类型、规格、已用/可用存储空间、分区上限、计费模式、可用区、实例ID、创建时间、内网连接地址、公网连接地址、企业项目和资源标签。

- 企业用户才能使用企业项目进行筛选。
- 资源标签筛选支持运算符操作，可选运算符包括：等于、不等于、包含、不包含。
- Kafka实例状态请参见[表9-1](#)。

表 9-1 Kafka 实例状态说明

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

| 状态   | 说明                                                                      |
|------|-------------------------------------------------------------------------|
| 创建失败 | Kafka实例处于创建失败状态。                                                        |
| 运行中  | Kafka实例正常运行状态。<br>在这个状态的实例可以运行您的业务。                                     |
| 故障   | Kafka实例处于故障状态。                                                          |
| 启动中  | Kafka实例从已冻结到运行中的中间状态。                                                   |
| 重启中  | Kafka实例正在进行重启操作。                                                        |
| 变更中  | Kafka实例正在进行规格变更、公网访问变更等操作。                                              |
| 变更失败 | Kafka实例处于规格变更、公网访问变更等操作失败的状态。<br>实例处于变更失败状态时，您无法重启实例、删除实例和修改实例，请联系客服处理。 |
| 已冻结  | Kafka实例处于已冻结状态。                                                         |
| 冻结中  | Kafka实例从运行中到已冻结的中间状态。                                                   |
| 升级中  | Kafka实例正在进行升级操作。                                                        |
| 回滚中  | Kafka实例正在进行回滚操作。                                                        |
| 回收中  | Kafka实例正在进行临时删除操作。                                                      |
| 回收站中 | Kafka实例处于回收站中。                                                          |
| 恢复中  | Kafka实例正在从回收站中恢复。                                                       |

**步骤4** 单击Kafka实例的名称，进入该Kafka实例的“概览”页面，查看Kafka实例的详细信息。

**表9-2**和**表9-3**为连接实例的相关参数，其他参数，请查看页面显示。

**表 9-2** 连接参数说明（不支持修改 SASL\_SSL）

| 信息类型 | 参数             | 说明                                    |
|------|----------------|---------------------------------------|
| 连接信息 | 用户名            | 开启SASL_SSL后，连接实例的用户名。                 |
|      | Kafka SASL_SSL | 是否开启SASL_SSL开关。<br>单机实例不支持开启SASL_SSL。 |
|      | 启用的安全协议        | 开启SASL_SSL后，实例已启用的安全协议。               |
|      | 开启的SASL认证机制    | 开启SASL_SSL后，实例已开启的SASL认证机制。           |
|      | SSL证书          | 单击“下载”，下载连接实例的SSL证书。                  |

| 信息类型 | 参数            | 说明                                                                           |
|------|---------------|------------------------------------------------------------------------------|
|      | 内网连接地址        | 未开启公网访问时，连接实例的地址。<br>显示的连接地址个数为实例代理个数。                                       |
|      | Manager内网访问地址 | 未开启公网访问时，连接Kafka Manager的地址。<br>2023年5月17日及以后创建的实例不提供此地址。                    |
|      | Manager用户名    | 连接Kafka Manager的用户名。<br>2023年5月17日及以后创建的实例不提供此用户名。                           |
|      | 公网访问          | 是否开启公网访问开关。                                                                  |
|      | 公网连接地址        | 开启公网访问后，连接实例的地址。<br>仅当开启了公网访问才显示该参数。                                         |
|      | Manager公网访问地址 | 开启公网访问后，连接Kafka Manager的地址。<br>仅当开启了公网访问才显示该参数。<br>2023年5月17日及以后创建的实例不提供此地址。 |
|      | VPC内网明文访问     | 是否开启VPC内网明文访问开关。                                                             |

表 9-3 连接参数说明（明文密文访问）

| 信息类型 | 参数   | 子参数      | 说明                             |
|------|------|----------|--------------------------------|
| 连接信息 | 用户名  | -        | 开启密文接入后，连接实例的用户名。              |
|      | 内网访问 | 明文接入     | 是否开启明文接入。                      |
|      |      | 内网明文连接地址 | 明文接入开启后，才显示此参数。                |
|      |      | 密文接入     | 是否开启密文接入。<br>单机实例不支持开启密文接入。    |
|      |      | 内网密文连接地址 | 密文接入开启后，才显示此参数。                |
|      |      | 启用的安全协议  | 密文接入开启后，才显示此参数。                |
|      | 公网访问 | 开关是否打开   | 是否开启公网访问。                      |
|      |      | 明文接入     | 公网访问开关打开后，才显示此参数。<br>是否开启明文接入。 |
|      |      | 公网明文连接地址 | 明文接入开启后，才显示此参数。                |

| 信息类型 | 参数          | 子参数      | 说明                                                      |
|------|-------------|----------|---------------------------------------------------------|
|      |             | 密文接入     | 公网访问开关打开后，才显示此参数。<br>是否开启密文接入。<br>单机实例不支持开启密文接入。        |
|      |             | 公网密文连接地址 | 密文接入开启后，才显示此参数。                                         |
|      |             | 启用的安全协议  | 密文接入开启后，才显示此参数。                                         |
|      | 开启的SASL认证机制 | -        | 密文接入开启后，才显示此参数。                                         |
|      | SSL证书       | -        | 仅在“启用的安全协议”为“SASL_SSL”时，才显示此参数。<br>单击“下载”，下载连接实例的SSL证书。 |

----结束

修改 Kafka 实例的基本信息

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例详情页面。
- 步骤4 以下参数支持修改。

表 9-4 Kafka 实例支持修改的参数

| 参数名称          | 修改方法                                                                                                                                                                                                                                                 | 查看修改结果                             |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| 实例名称          | 单击  ，输入修改后的实例名称，单击  。<br>实例名称需要符合命名规则：长度为4~64个字符，由英文字母开头，只能由英文字母、数字、中划线、下划线组成。 | 页面上方直接提示修改结果。                      |
| Smart Connect | 修改方法请参考 <a href="#">开启Smart Connect</a> 和 <a href="#">关闭Smart Connect</a> 。                                                                                                                                                                          | 在“实例管理 > 后台任务管理”页面，查看当前任务的操作进度和结果。 |
| 描述            | 单击  ，输入修改后的描述，单击  。<br>长度为0~1024个字符。                                           | 页面上方直接提示修改结果。                      |

| 参数名称           | 修改方法                                                                                                                                                                                                                     | 查看修改结果                           |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------|
| 企业项目           | 单击  ，在下拉框中选择新的企业项目，单击  。<br>该参数针对企业用户使用。修改企业项目，不会重启实例。 | 页面上方直接提示修改结果。                    |
| 安全组            | 单击  ，在下拉框中选择新的安全组，单击  。<br>修改安全组，不会重启实例。               | 页面上方直接提示修改结果。                    |
| 内网访问           | 修改方法请参考 <a href="#">配置Kafka实例的明文/密文接入</a> 。                                                                                                                                                                              | 系统跳转到“后台任务管理”页面，并显示当前任务的操作进度和结果。 |
| 公网访问           | 修改方法请参考 <a href="#">配置Kafka实例的公网访问</a> 。                                                                                                                                                                                 | 系统跳转到“后台任务管理”页面，并显示当前任务的操作进度和结果。 |
| 容量阈值策略         | 单击待选择的容量阈值策略，弹出“确认”对话框，单击“确定”。<br>修改容量阈值策略，不会重启实例。<br>磁盘使用量达到容量阈值时会触发对应的阈值处理策略，可能会导致消息被删除或无法继续生产消息。                                                                                                                      | 系统跳转到“后台任务管理”页面，并显示当前任务的操作进度和结果。 |
| Kafka自动创建Topic | 开启/关闭Kafka自动创建Topic，弹出“确认”对话框，单击“确定”。<br>开启/关闭“Kafka自动创建Topic”可能会导致实例重启，具体以控制台显示为准。                                                                                                                                      | 系统跳转到“后台任务管理”页面，并显示当前任务的操作进度和结果。 |
| Kafka自动创建消费组   | 修改方法请参考 <a href="#">配置Kafka自动创建消费组</a> 。                                                                                                                                                                                 | 系统跳转到“后台任务管理”页面，并显示当前任务的操作进度和结果。 |
| 自动磁盘扩容         | 修改方法请参考 <a href="#">自动扩容Kafka实例存储空间</a> 。                                                                                                                                                                                | 页面上方直接提示修改结果。                    |
| 跨VPC访问         | 修改方法请参考 <a href="#">使用VPCEP实现跨VPC访问Kafka</a> 和 <a href="#">使用DNAT实现公网访问Kafka</a> 。                                                                                                                                       | 页面上方直接提示修改结果。                    |

----结束

相关文档

- 如果您希望通过API查询Kafka实例信息，请参考[查询指定实例](#)。
- 如果您希望通过API修改Kafka实例基本信息，请参考[修改实例信息](#)。

## 9.2 查看 Kafka 磁盘使用量

本章节指导您在控制台查看每个代理上磁盘的使用量，以了解Kafka实例的磁盘使用情况。

### 约束与限制

单机实例不支持查看磁盘使用量。

### 查看 Kafka 磁盘使用量

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例的名称，进入实例的“概览”页面。
- 步骤4 在“监控 > 磁盘存储统计”页面，查看磁盘的使用量。

页面支持从以下三个方面对代理中的Topic磁盘使用量进行查询：使用量排名前xx个、使用量达到xx GB和使用占比达xx%。

在页面右上角，单击“查看监控数据”，跳转到云监控服务的监控指标页面。在监控指标页面，您可以查看Kafka实例的各项监控指标。

图 9-1 查看磁盘的使用量



----结束

### 相关文档

如果您希望通过API查看Topic磁盘使用量，请参考[查询Topic的磁盘存储情况](#)。

### 9.3 查看 Kafka 实例后台任务

对实例的一些操作，如[表9-5](#)所示，会启动一个后台任务，您可以在后台任务页，查看该操作的状态等信息，同时可通过删除操作，清理任务信息。

表 9-5 后台任务列表

| 任务名             | 说明                                                                                                                           |
|-----------------|------------------------------------------------------------------------------------------------------------------------------|
| 创建实例            | 创建一个Kafka实例。                                                                                                                 |
| 重启实例            | 重启Kafka实例。                                                                                                                   |
| Kafka配置参数修改     | <ul style="list-style-type: none"><li>• 修改Kafka的配置参数。</li><li>• 开启/关闭自动创建Topic。</li></ul>                                    |
| 容量阈值策略变更        | 修改Kafka实例的容量阈值策略。                                                                                                            |
| Kafka动态配置SSL    | 修改明文接入/密文接入。                                                                                                                 |
| 配置公网访问          | 开启/关闭公网访问。                                                                                                                   |
| 开启Smart Connect | 开启Smart Connect功能。                                                                                                           |
| 关闭Smart Connect | 关闭Smart Connect功能。                                                                                                           |
| 变更规格            | <ul style="list-style-type: none"><li>• 扩容存储空间</li><li>• 扩容代理数量</li><li>• 扩容基准带宽</li><li>• 扩容代理规格</li><li>• 缩容代理规格</li></ul> |
| 创建流控            | 创建用户/客户端/Topic流控。                                                                                                            |
| 编辑流控            | 修改流控参数。                                                                                                                      |
| 删除流控            | 删除用户/客户端/Topic流控。                                                                                                            |
| Kafka分区平衡       | 修改Topic的分区平衡。                                                                                                                |
| 开启日志            | 开启Kafka重平衡日志。                                                                                                                |
| 关闭日志            | 关闭Kafka重平衡日志。                                                                                                                |
| 设置Topic权限       | 在Topic中为不同的用户设置不同的权限。                                                                                                        |

#### 查看 Kafka 实例后台任务

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例的“概览”页面。

**步骤4** 在左侧导航栏单击“实例管理 > 后台任务管理”，进入后台任务管理页面。

**步骤5** 在“即时任务”/“定时任务”页签，单击时间段下拉框，选择搜索时间，在搜索框中输入关键字，按“Enter”，界面显示相应时间段内启动的任务。

在后台任务管理页面，您还可以执行以下操作：

- 单击，刷新任务状态。
- 单击“删除”，弹出“删除任务记录”对话框，单击“确定”，清理任务信息。  
只有在任务已经执行完成，即任务状态为成功或者失败，或者定时任务已经取消，即任务状态为取消，才能执行删除操作。

----结束

## 相关文档

如果您希望通过API查看后台任务列表，请参考[查询实例的后台任务列表](#)。

## 9.4 查看 Kafka 生产消费示例代码

分布式消息服务Kafka版支持在控制台查看Java、Go和Python语言的生产消费消息示例代码，您可以参考示例代码快速完成Kafka客户端的集成，并进行消息的生产和消费。

### 查看 Kafka 生产消费示例代码

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入实例详情页面。

**步骤4** 在左侧导航栏选择“实例管理 > Topic管理”，进入Topic列表页面。

**步骤5** 单击“查看示例代码”，弹出“生产消费示例代码”对话框。

示例代码根据语言分为：Java、Go和Python。根据接入类型分为：PlainText、SASL\_SSL和SASL\_PLAINTEXT。

- PlainText：客户端通过明文连接Kafka实例。
- SASL\_SSL：客户端通过密文连接Kafka实例，且连接时需要进行SASL认证。SASL认证分为PLAIN和SCRAM-SHA-512。
- SASL\_PLAINTEXT：客户端通过明文连接Kafka实例，且连接时需要进行SASL认证。SASL认证分为PLAIN和SCRAM-SHA-512。

----结束

## 9.5 修改 Kafka 实例配置参数

分布式消息服务Kafka版为实例、Topic、消费者提供了几个常用配置参数的默认值，您可以根据实际业务需求，在控制台自行修改参数值。其他在控制台未列出的配置参数，请参考[Kafka配置](#)进行修改。

Kafka实例的参数分为动态参数和静态参数：

- 动态参数：动态参数修改成功后，无需重启实例，立即生效。
- 静态参数：静态参数修改成功后，需要手动重启实例才能生效。

约束与限制

- 部分老实例不支持修改配置参数，具体以控制台为准，请联系客服解决。
- 单机实例不支持修改配置参数。

前提条件

Kafka实例的状态为“运行中”时，才能修改配置参数。

修改 Kafka 实例配置参数

步骤1 登录[Kafka控制台](#)。

步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。

步骤3 单击Kafka实例的名称，进入实例详情页面。

步骤4 在“实例管理 > 配置参数”页面，修改配置参数。

- 修改指定参数：在待修改参数所在行，单击“编辑”。
- 批量修改参数：单击“批量编辑”。
- 恢复参数默认值：单击“编辑”/“批量编辑”，然后单击“恢复默认”。

1.1.0版本实例的参数说明如[表9-6](#)和[表9-7](#)所示，2.3.0/2.7/3.x版本实例的参数说明如[表9-8](#)和[表9-9](#)所示。

表 9-6 动态参数说明（1.1.0 版本）

| 参数                        | 参数说明                                                                                                                     | 参数范围                  | 默认值   |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------|-----------------------|-------|
| auto.create.groups.enable | 是否开启自动创建消费组功能。<br>2023年4月25日及以后创建的实例支持在控制台修改此参数，在此之前创建的实例默认开启自动创建消费组功能，无法在控制台关闭此功能。                                      | true/false            | true  |
| offsets.retention.minutes | 消费位点保留时间是消费位点最大保留时间，从提交消费位点的时间开始计算，超过该时间的消费位点将被删除。消费者每次向Topic分区提交一次消费位点时，该消费位点的保留时间将会被重置为0。<br>2023年5月1日前创建的实例，此参数为静态参数。 | 1440 ~ 30240<br>单位：分钟 | 20160 |

表 9-7 静态参数说明（1.1.0 版本）

| 参数                             | 参数说明                                                                                                                                                                                                                                        | 参数范围                    | 默认值      |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|----------|
| min.insync.replicas            | 当producer将acks设置为“all”（或“-1”）时，此配置指定必须确认写入才能被认为成功的副本的最小数量。                                                                                                                                                                                  | 1 ~ 3                   | 1        |
| message.max.bytes              | 单条消息的最大长度。                                                                                                                                                                                                                                  | 0 ~ 10485760<br>单位：字节   | 10485760 |
| unclean.leader.election.enable | 指示是否启用不在ISR集合中的副本选为领导者作为最后的手段，即使这样做可能导致数据丢失。                                                                                                                                                                                                | true/false              | false    |
| connections.max.idle.ms        | 此参数用来指定在多少毫秒之后，关闭空闲的连接。                                                                                                                                                                                                                     | 5000 ~ 600000<br>单位：毫秒  | 600000   |
| log.retention.hours            | 实例级消息老化时间（小时为单位）。<br>如果Topic已经设置了老化时间，则此参数对此Topic不生效。仅在Topic未设置老化时间时，此参数才对此Topic生效。                                                                                                                                                         | 1 ~ 168<br>单位：小时        | 72       |
| max.connections.per.ip         | 每个IP允许的最大连接数。超过此连接数的连接请求将被丢弃。                                                                                                                                                                                                               | 100 ~ 20000             | 1000     |
| group.max.session.timeout.ms   | consumer允许的最大会话超时时间。超时时间越长，consumer就能在心跳探测周期内有更多时间处理消息，但也会使故障检测花费更长时间。                                                                                                                                                                      | 6000 ~ 1800000<br>单位：毫秒 | 1800000  |
| default.replication.factor     | 自动创建Topic时的默认副本个数。                                                                                                                                                                                                                          | 1 ~ 3                   | 3        |
| allow.everyone.if.no.acl.found | <ul style="list-style-type: none"> <li>如果设置为true，没有给资源配置ACL时，允许所有用户访问。</li> <li>如果设置为false，初始用户拥有所有操作的权限，其他用户需要配置权限才能进行对应的操作。所有操作包括修改Topic权限、创建和删除Topic、以及修改Topic分区数。</li> </ul> <p>仅在实例开启密文接入时，存在此参数。</p> <p>2023年9月15日前创建的实例不支持修改此参数。</p> | true/false              | true     |

| 参数                           | 参数说明                                                                      | 参数范围                   | 默认值  |
|------------------------------|---------------------------------------------------------------------------|------------------------|------|
| num.partitions               | 自动创建Topic时的默认分区数。                                                         | 1 ~ 200                | 3    |
| group.min.session.timeout.ms | consumer允许的最小会话超时时间。超时时间越短，consumer的心跳探测越频繁，可以使故障检测更快，但会导致broker被抢占更多的资源。 | 6000 ~ 300000<br>单位：毫秒 | 6000 |

表 9-8 动态参数说明（2.3.0/2.7/3.x 版本）

| 参数                             | 参数说明                                                                                                                     | 参数范围                  | 默认值      |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------|-----------------------|----------|
| min.insync.replicas            | 当producer将acks设置为“all”（或“-1”）时，此配置指定必须确认写入才能被认为成功的副本的最小数量。                                                               | 1 ~ 3                 | 1        |
| message.max.bytes              | 单条消息的最大长度。                                                                                                               | 0 ~ 10485760<br>单位：字节 | 10485760 |
| auto.create.groups.enable      | 是否开启自动创建消费组功能。<br>2023年4月25日及以后创建的实例支持在控制台修改此参数，在此之前创建的实例默认开启自动创建消费组功能，无法在控制台关闭此功能。                                      | true/false            | true     |
| max.connections.per.ip         | 每个IP允许的最大连接数。超过此连接数的连接请求将被丢弃。                                                                                            | 100 ~ 20000           | 1000     |
| unclean.leader.election.enable | 指示是否启用不在ISR集合中的副本选为领导者作为最后的手段，即使这样做可能导致数据丢失。                                                                             | true/false            | false    |
| offsets.retention.minutes      | 消费位点保留时间是消费位点最大保留时间，从提交消费位点的时间开始计算，超过该时间的消费位点将被删除。消费者每次向Topic分区提交一次消费位点时，该消费位点的保留时间将会被重置为0。<br>2023年5月1日前创建的实例，此参数为静态参数。 | 1440 ~ 30240<br>单位：分钟 | 20160    |

表 9-9 静态参数说明（2.3.0/2.7/3.x 版本）

| 参数                             | 参数说明                                                                                                                                                                                                                                     | 参数范围                    | 默认值     |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|---------|
| connections.max.idle.ms        | 此参数用来指定在多少毫秒之后，关闭空闲的连接。                                                                                                                                                                                                                  | 5000 ~ 600000<br>单位：毫秒  | 600000  |
| log.retention.hours            | 实例级消息老化时间（小时为单位）<br>如果Topic已经设置了老化时间，则此参数对此Topic不生效。仅在Topic未设置老化时间时，此参数才对此Topic生效。                                                                                                                                                       | 1 ~ 168<br>单位：小时        | 72      |
| group.max.session.timeout.ms   | consumer允许的最大会话超时时间。超时时间越长，consumer就能在心跳探测周期内有更多时间处理消息，但也会使故障检测花费更长时间。                                                                                                                                                                   | 6000 ~ 1800000<br>单位：毫秒 | 1800000 |
| default.replication.factor     | 自动创建Topic时的默认副本个数。                                                                                                                                                                                                                       | 1 ~ 3                   | 3       |
| allow.everyone.if.no.acl.found | <ul style="list-style-type: none"> <li>如果设置为true，没有给资源配置ACL时，允许所有用户访问。</li> <li>如果设置为false，初始用户拥有所有操作的权限，其他用户需要配置权限才能进行对应的操作。所有操作包括修改Topic权限、创建和删除Topic、以及修改Topic分区数。</li> </ul> <p>仅在实例开启密文接入时，存在此参数。<br/>2023年9月15日前创建的实例不支持修改此参数。</p> | true/false              | true    |
| num.partitions                 | 自动创建Topic时的默认分区数。                                                                                                                                                                                                                        | 1 ~ 200                 | 3       |
| group.min.session.timeout.ms   | consumer允许的最小会话超时时间。超时时间越短，consumer的心跳探测越频繁，可以使故障检测更快，但会导致broker被抢占更多的资源。                                                                                                                                                                | 6000 ~ 300000<br>单位：毫秒  | 6000    |

**步骤5** 单击“保存”，完成参数的修改。

动态参数修改成功后，无需重启实例，立即生效。**静态参数修改成功后，需要手动重启实例才能生效。**

在“配置参数”页面的“参数运行值”列，查看修改后的参数值。

----结束

相关文档

如果您希望通过API修改Kafka实例的配置参数，请参考[修改实例配置](#)。

9.6 配置 Kafka 实例标签

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

如您的组织已经设定分布式消息服务Kafka版的相关标签策略，则需按照标签策略规则为Kafka实例添加标签。新添加的标签如果不符合标签策略的规则，会添加失败。

标签共由两部分组成：“标签键”和“标签值”，其中，“标签键”和“标签值”的命名规则如[表9-10](#)所示。

表 9-10 标签命名规则

| 参数名称 | 规则                                                                                                                                                                                                      |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 标签键  | <ul style="list-style-type: none"><li>不能为空。</li><li>对于同一个实例，Key值唯一。</li><li>长度为1~128个字符（中文也可以输入128个字符）。</li><li>由任意语种字母、数字、空格和字符组成，字符仅支持 _ . : = + - @</li><li>首尾字符不能为空格。</li><li>不能以_sys_开头。</li></ul> |
| 标签值  | <ul style="list-style-type: none"><li>长度为0~255个字符（中文也可以输入255个字符）。</li><li>由任意语种字母、数字、空格和字符组成，字符仅支持 _ . : = + - @</li><li>创建实例时设置的标签值，首尾字符不能为空格。</li></ul>                                               |

约束限制

一个Kafka实例最多添加20个标签。

配置单个 Kafka 实例标签

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 通过以下任意一种方法，配置Kafka实例标签。
- 方法一：
  - 单击实例名称，进入实例详情页面。
  - 在左侧导航栏单击“实例管理 > 标签”，进入标签列表页面。
  - 单击“编辑标签”，弹出“编辑标签”对话框。

4. 您可以根据实际需要，添加/删除/编辑标签。

表 9-11 标签操作说明

| 操作类型 | 操作步骤                                                                                                                                            |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| 添加标签 | 1. 单击“添加标签”，在“标签键”和“标签值”中，输入标签的键/值。如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键/值。<br>最多可以添加 <b>20个标签</b> 。<br>2. 单击“确定”，成功为实例添加标签。<br>在标签列表页面，查看新添加的标签。 |
| 编辑标签 | 修改标签键/值，单击“确定”，完成标签的修改。<br>在标签列表页面，查看修改后的标签键/值。                                                                                                 |
| 删除标签 | 在待删除的标签所在行，单击“删除”。然后单击“确定”，完成标签的删除。<br>已删除的标签不再显示在标签列表中，表明标签已成功删除。                                                                              |

方法二：

1. 在待配置标签的Kafka实例所在行，单击“更多 > 编辑标签”，弹出“编辑标签”对话框。
2. 您可以根据实际需要，添加/删除/编辑标签。

表 9-12 标签操作说明

| 操作类型 | 操作步骤                                                                                                                                                                                   |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 添加标签 | 1. 单击“添加标签”，在“标签键”和“标签值”中，输入标签的键/值。如果您已经预定义了标签，在“标签键”和“标签值”中选择已经定义的标签键/值。<br>最多可以添加 <b>20个标签</b> 。<br>2. 单击“确定”，成功为实例添加标签。<br>3. 在已添加标签的Kafka实例所在行，单击“更多 > 编辑标签”，弹出“编辑标签”对话框。查看新添加的标签。 |
| 编辑标签 | 1. 修改标签键/值，单击“确定”，完成标签的修改。<br>2. 在已修改标签的Kafka实例所在行，单击“更多 > 编辑标签”，弹出“编辑标签”对话框。查看修改后的标签键/值。                                                                                              |

| 操作类型 | 操作步骤                                                                                                                                                                   |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 删除标签 | <ol style="list-style-type: none"><li>1. 在待删除的标签所在行，单击“删除”。然后单击“确定”，完成标签的删除。</li><li>2. 在已删除标签的Kafka实例所在行，单击“更多 &gt; 编辑标签”，弹出“编辑标签”对话框。已删除的标签不再显示，表明标签已成功删除。</li></ol> |

----结束

## 相关文档

如果您希望通过API添加或删除Kafka实例标签，请参考[批量添加或删除实例标签](#)。

## 9.7 配置 Kafka 实例回收站策略

开启回收站策略的实例在被删除后，会被临时存入回收站中，此时实例中的数据尚未被彻底删除，在保留天数内支持从回收站中恢复此实例。超过保留天数的实例会被彻底删除，无法恢复。

回收站策略默认是关闭状态。

### 约束与限制

- 回收站中的按需实例不会收取实例的费用，但是会收取存储空间的费用。
- 包年/包月的实例退订后会存入回收站中，此时不会收取实例的费用，但是会按需收取存储空间的费用。
- 包年/包月的实例从回收站中恢复后，计费模式会变为按需计费。
- 处于宽限期和保留期的实例，删除/退订后，实例不会进入回收站中，会被彻底删除。

### 开启回收站策略

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 单击“回收站策略”，弹出“回收站策略”对话框。

**步骤5** 开启启用回收站，并设置保留天数（只能设置为1~7天），单击“确定”。

----结束

### 恢复 Kafka 实例

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 通过以下任意一种方法，恢复Kafka实例。

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

**步骤5** 在弹出的“恢复实例”对话框中，单击“确定”。

恢复实例大约需要3~10分钟，在“Kafka实例”页面查看已恢复的实例。

包年/包月的实例从回收站中恢复后，计费模式会变为按需计费。

----结束

## 修改保留天数

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 单击“回收站策略”，弹出“回收站策略”对话框。

**步骤5** 修改保留天数（只能设置为1~7天），单击“确定”。

修改保留天数仅对修改后删除的实例生效，修改前已在回收站中的实例依然按照修改前的天数保留实例。

----结束

## 导出回收站中的实例

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 通过以下任意一种方法，导出实例列表。

- 勾选待导出的实例，单击“导出 > 导出已选中数据到XLSX”，导出指定实例列表。
- 单击“导出 > 导出全部数据到XLSX”，导出全部实例列表。

----结束

## 删除回收站中的实例

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 通过以下任意一种方法，删除实例。

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

**步骤5** 在弹出的“删除实例”对话框中，输入“DELETE”，单击“确定”，完成删除Kafka实例。

回收站中的Kafka实例删除后，实例中原有的数据将被删除，且没有备份，请谨慎操作。

----结束

## 关闭回收站策略

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 在左侧导航栏选择“回收站”，进入“回收站”页面。

**步骤4** 单击“回收站策略”，弹出“回收站策略”对话框。

**步骤5** 关闭启用回收站，单击“确定”。

----结束

## 9.8 升级 Kafka 实例内核版本

升级Kafka实例内核版本指的是将老的内核版本升级到最新的内核版本，升级后，Kafka实例将支持一些新的特性（例如支持在控制台创建消费组、支持查看Topic详情等），并且修复老版本的一些问题。

升级Kafka实例内核版本并非升级Kafka实例的版本，例如升级前Kafka版本为2.7，升级内核版本后，Kafka版本还是2.7。

### 升级内核版本的影响

- 若Topic为单副本，升级期间无法对该Topic生产消息或消费消息，会造成业务中断。
- 若Topic为多副本，升级不会造成服务中断，但会逐个节点重启，负载会转移到剩余节点上，建议您在业务低峰期升级。
- 升级过程中会逐个节点升级，单个节点的升级包括两部分：升级软件包和数据同步。升级软件包耗时在5分钟左右，数据同步耗时取决于升级软件包过程中其他节点Leader副本的生产数据量，数据量越大，所需时间越久。升级总耗时=每个节点升级软件包耗时+数据同步耗时。
- 升级过程中会逐个节点重启监控进程，导致监控数据断点，重启成功后，监控数据恢复。
- 升级过程中节点滚动重启造成分区Leader切换，会发生秒级连接闪断，在用户网络环境稳定的前提下，Leader切换时长一般为1分钟以内。多副本的Topic需要在生产客户端配置重试机制，方法如下：
  - 生产客户端为Kafka开源客户端时，检查是否配置retries和retry.backoff.ms参数。建议参数值分别配置为：retries=10，retry.backoff.ms=1000。

- 生产客户端为Flink客户端时，检查是否配置重启策略，配置重启策略可以参考如下代码。  
StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvironment();  
env.setRestartStrategy(RestartStrategies.fixedDelayRestart(10, Time.seconds(20)));

前提条件

Kafka实例的状态为“运行中”时，才能升级Kafka实例内核版本。

升级 Kafka 实例内核版本

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 通过以下任意一种方法，升级Kafka实例。
  - 在待升级内核版本的Kafka实例所在行，单击“更多 > 升级”，弹出“版本升级”对话框。
  - 单击待升级内核版本的Kafka实例名称，进入实例详情页面。单击右上角的“... > 升级”，弹出“版本升级”对话框。
- 步骤4 选择执行时间，支持“立即执行”和“定时执行”。
- 如果选择“定时执行”，还需要指定执行时间。
- 步骤5 在“风险检查”区域，查看升级前的检查项是否正常。
- 如果检查项有异常，请先根据提示信息修改，然后单击“重新检查”。如果确认异常的检查项无需修改，请勾选“我已知晓风险。”。
- 步骤6 单击“确定”。
- 立即执行和定时执行查看升级是否完成的方法不同，具体如下：

表 9-13 查看升级结果

| 执行方式 | 升级结果                                                                                                                                                                                                                                    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 立即执行 | <div>1. 单击升级实例名称，进入实例详情页。</div> <div>2. 在左侧导航栏单击“实例管理 &gt; 后台任务管理”，进入“后台任务管理”页面。</div> <div>3. 在“即时任务”页签中，查看升级任务的状态。<ul style="list-style-type: none"><li>当任务的“状态”为“成功”时，表示升级完成。</li><li>当任务的“状态”为“失败”时，表示升级失败，请联系客服处理。</li></ul></div> |

| 执行方式 | 升级结果                                                                                                                                                                                                                                                                                                                                                                                                                      |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 定时执行 | <ol style="list-style-type: none"><li>1. 单击升级实例名称，进入实例详情页。</li><li>2. 在左侧导航栏单击“实例管理 &gt; 后台任务管理”，进入“后台任务管理”页面。</li><li>3. 在“定时任务”页签中，查看升级任务是否开始执行。<ul style="list-style-type: none"><li>- 当“状态”为“待执行”时，表示升级任务未执行。</li><li>- 当“状态”为“成功”时，表示升级任务开始执行。</li></ul></li><li>4. 在“即时任务”页签中，查看升级任务的状态。<ul style="list-style-type: none"><li>- 当任务的“状态”为“成功”时，表示升级完成。</li><li>- 当任务的“状态”为“失败”时，表示升级失败，请联系客服处理。</li></ul></li></ol> |

----结束

### 修改定时升级任务

- 步骤1** 在“实例管理 > 后台任务管理”页面的“定时任务”页签中，单击页面左上角下拉框，选择时间段，在搜索对话框中输入“版本升级”，按“Enter”，快速查找定时升级任务。
- 步骤2** 在待修改的定时升级任务后，单击“修改”。
- 步骤3** 在弹出的“修改定时任务”对话框中，您可以修改定时升级任务的时间，还可以取消定时升级任务，具体操作如下。
- 修改定时升级任务的时间：修改时间，单击“确定”。
  - 取消定时升级任务：选择“取消”，单击“确定”。

----结束

### 相关文档

如果您希望通过API升级Kafka实例内核版本，请参考[实例升级](#)。

## 9.9 导出 Kafka 实例列表

分布式消息服务Kafka版支持在控制台导出实例列表信息，供用户查看和使用。

### 导出 Kafka 实例列表

- 步骤1** 登录[Kafka控制台](#)。
- 步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3** 通过以下任意一种方法，导出实例列表。
- 勾选待导出的实例，单击“导出 > 导出已选中数据到XLSX”，导出指定实例列表。

- 单击“导出 > 导出全部数据到XLSX”，导出全部实例列表。

----结束

## 9.10 重启 Kafka 实例

分布式消息服务Kafka版支持在控制台重启单个Kafka实例，或批量重启Kafka实例。

### 约束与限制

- 在Kafka实例重启过程中，客户端的生产与消费消息等请求会被拒绝。
- 在客户端配置重试机制，避免因为重启实例导致的业务断连。

### 前提条件

只有当Kafka实例处于“运行中”或“故障”状态，才能执行重启操作。

### 重启 Kafka 实例

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，重启Kafka实例。

- 勾选Kafka实例名称左侧的方框，可选一个或多个，单击信息栏左上侧的“重启”。
- 在待重启Kafka实例所在行，单击“更多 > 重启”。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 重启”。

**步骤4** 在“重启实例”对话框中，单击“确定”，完成重启Kafka实例。

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

#### 说明

重启Kafka实例只会重启实例进程，不会重启实例所在虚拟机。

----结束

### 相关文档

如果您希望通过API重启Kafka实例，请参考[批量重启或删除实例](#)。

## 9.11 删除 Kafka 实例

对于按需计费的Kafka实例，控制台支持删除单个实例，也可批量删除实例。对于包年/包月的Kafka实例，如果不再使用，单击“操作”栏下的“更多 > 退订”，退订成功后，Kafka实例会自动被删除。

删除的实例能否恢复取决于是否开启回收站策略，如果未开启回收站策略，实例删除后，实例中原有的数据将被删除，且没有备份。回收站策略默认关闭，如果需要开启，具体步骤请参见[开启回收站策略](#)。

## 前提条件

Kafka实例状态为运行中、故障、已冻结、创建失败的按需付费实例才能执行删除操作。

## 删除 Kafka 实例

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，删除按需计费的Kafka实例。

- 勾选Kafka实例名称左侧的方框，可选一个或多个，单击信息栏左上侧的“删除”。
- 在待删除Kafka实例所在行，单击“更多 > 删除”。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 删除”。

**步骤4** 在“删除实例”对话框中，输入“DELETE”，单击“确定”，完成删除Kafka实例。

删除Kafka实例大约需要1到60秒。

已删除的Kafka实例不再显示在实例列表中，表明Kafka实例已成功删除。

----结束

## 相关文档

- 如果您希望通过API删除指定Kafka实例，请参考[删除指定的实例](#)。
- 如果您希望通过API批量删除Kafka实例，请参考[批量重启或删除实例](#)。

# 9.12 管理 Kafka Manager

## 9.12.1 访问 Kafka Manager

Kafka Manager是开源的Kafka集群管理工具，需要通过浏览器才能访问Kafka Manager的地址。在Kafka Manager页面，您可以查看Kafka集群的监控、代理等信息。

2023年5月17日及以后创建的实例不提供Kafka Manager，Kafka实例控制台已支持Kafka Manager相关功能，具体如下：

表 9-14 Kafka Manager 功能对应的 Kafka 实例控制台功能

| Kafka Manager功能 | Kafka实例控制台功能                   |
|-----------------|--------------------------------|
| 查看实例的Topic      | 在“实例管理 > Topic管理”页面，查看Topic列表。 |

| Kafka Manager功能 | Kafka实例控制台功能                                                                             |
|-----------------|------------------------------------------------------------------------------------------|
| 查看队列基本信息        | 在“实例管理 > Topic管理”页面，查看每个Topic的基本信息，包括副本数、分区数和老化时间等。                                      |
| Topic分区平衡       | 在“实例管理 > Topic管理”页面，修改分区平衡，支持手动平衡和自动平衡。                                                  |
| 更新Topic配置       | 在“实例管理 > Topic管理”页面，修改Topic配置参数。                                                         |
| 查看消费组列表         | 在“实例管理 > 消费组管理”页面，查看消费组列表。                                                               |
| 查看消费组详情页        | 在“实例管理 > 消费组管理”页面，单击消费组名称，进入消费组详情页，查看消费者列表和消费进度。                                         |
| 查看消费组队列详情页      | 在“实例管理 > 消费组管理”页面，单击消费组名称，进入消费组详情页。在“消费进度”页签中，查看消费组消费的所有Topic列表以及每个Topic的总堆积数和每个分区的消费状态。 |
| 集群监控信息/队列监控信息   | 在“监控 > 监控详情”页面，查看实例的监控信息。                                                                |

前提条件

已配置如表9-15所示安全组。

表 9-15 安全组规则

| 方向  | 协议  | 端口   | 源地址                 | 说明               |
|-----|-----|------|---------------------|------------------|
| 入方向 | TCP | 9999 | Kafka客户端所在的IP地址或地址组 | 访问Kafka Manager。 |

登录 Kafka Manager

- 步骤1 创建一台与Kafka实例相同VPC和相同安全组的Windows服务器，详细步骤请参考[购买并登录Windows弹性云服务器](#)。
- 如果是已经开启了公网访问，该步骤为可选，在本地浏览器中即可访问，不需要单独的Windows弹性云服务器。
- 步骤2 在实例详情信息页面，获取Kafka Manager地址。
  - 未开启公网访问时，Kafka Manager地址为“Manager内网访问地址”。

图 9-2 Kafka Manager 内网访问地址



- 已开启公网访问时，Kafka Manager地址为“Manager公网访问地址”。

图 9-3 Kafka Manager 公网访问地址

Manager公网访问地址 <https://122.150:9999>, <https://122.136:9999>

**步骤3** 在浏览器中输入Kafka Manager的地址，进入Kafka Manager登录页面。

如果是开启了公网访问，在本地浏览器输入Kafka Manager地址访问；如果没有开启公网访问，需要登录**步骤1**的弹性云服务器，然后在浏览器输入Kafka Manager地址访问。

**步骤4** 输入创建实例时设置的Kafka Manager用户名和密码，即可管理Kafka集群。

----结束

## 查看 Kafka Manager

在进入Kafka Manager集群管理页面后，您可以查看Kafka集群的监控、代理等信息。

- 集群信息页

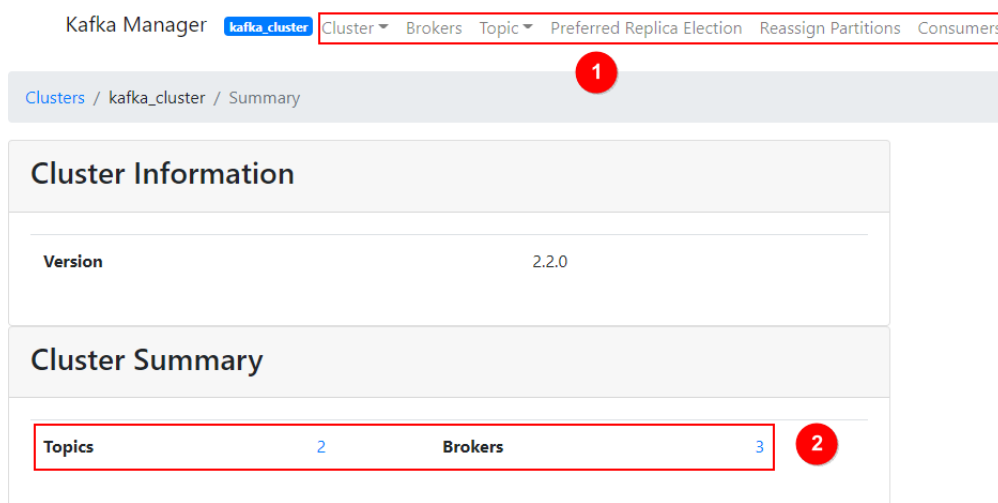
单击**Clusters**中的集群列表，即可进入集群信息页。如图9-4所示。

– 图中①区域表示功能导航栏。

- Cluster：集群，统计集群列表和集群详情。
- Brokers：代理，统计当前集群中各代理的状态信息。
- Topic：队列，统计当前集群中的kafka队列。
- Preferred Replica Election：强制进行一次队列leader的最优选举（不建议用户操作）。
- Reassign Partitions：进行分区副本的重分配（不建议用户操作）。
- Consumers：统计集群中的消费组状态。

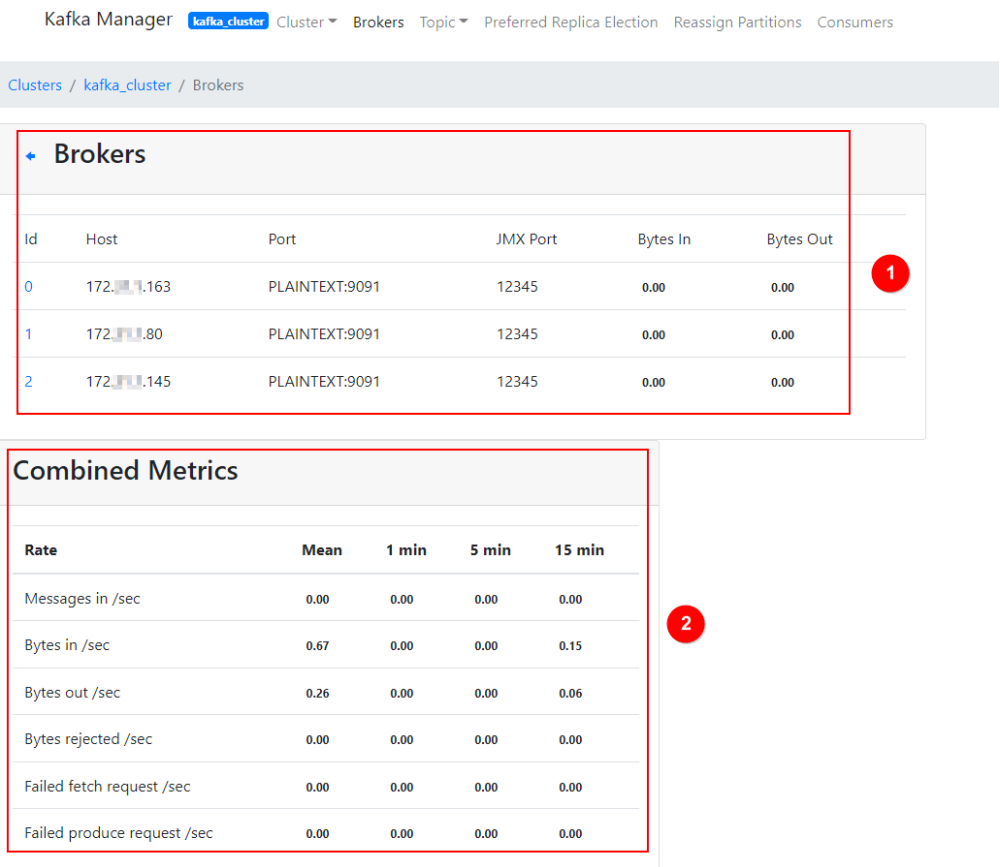
– 图中②区域表示集群信息统计，包含集群的Topic数和集群的代理数。

图 9-4 集群信息页



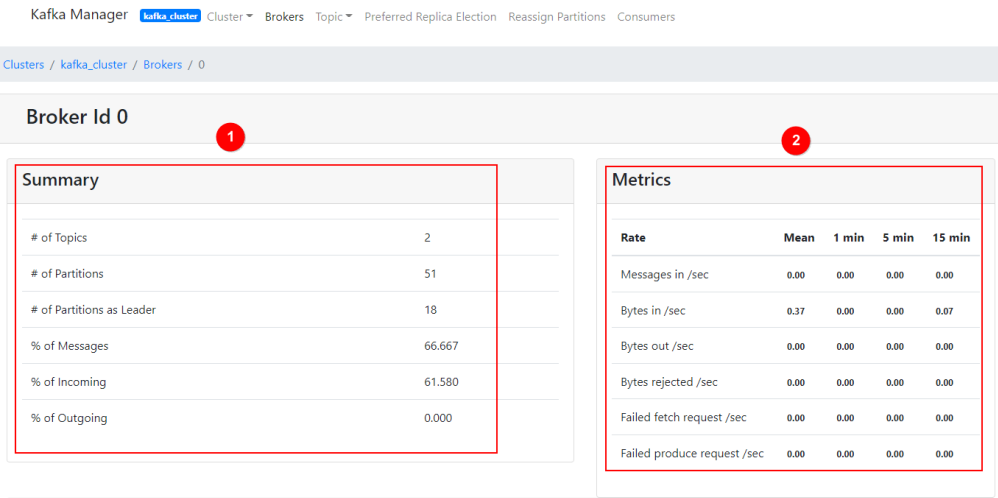
- 集群所有代理统计页  
单击功能导航栏中的**Brokers**，即可进入代理统计页。如图9-5所示。
  - 图中①区域代理列表，包含总的字节流入和字节流出。
  - 图中②集群监控信息。

图 9-5 所有代理统计页



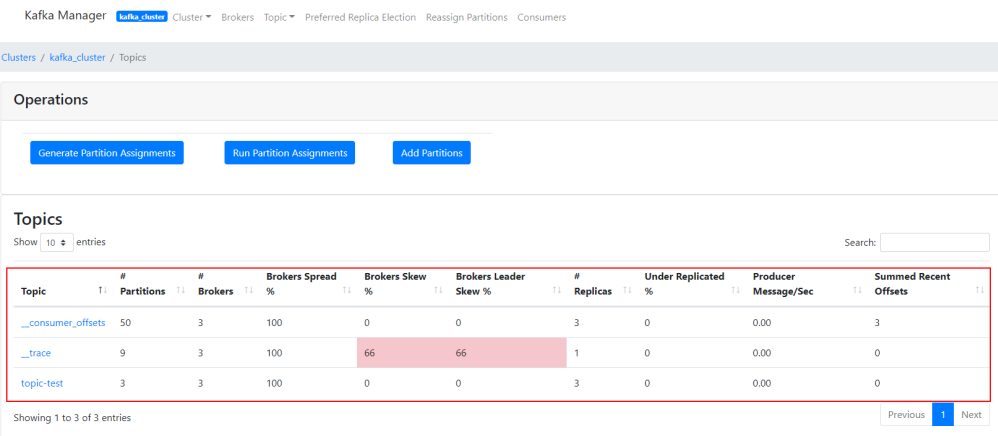
- 具体代理统计页  
单击id列表中具体的Broker，即可查看对应代理的统计信息。如图9-6所示。
  - 图中①区域表示对应代理总的统计信息，包括队列数、分区数、分区leader数、消息速率占比、写入字节占比以及流出字节占比。
  - 图中②区域表示代理监控信息。

图 9-6 具体 Broker 信息



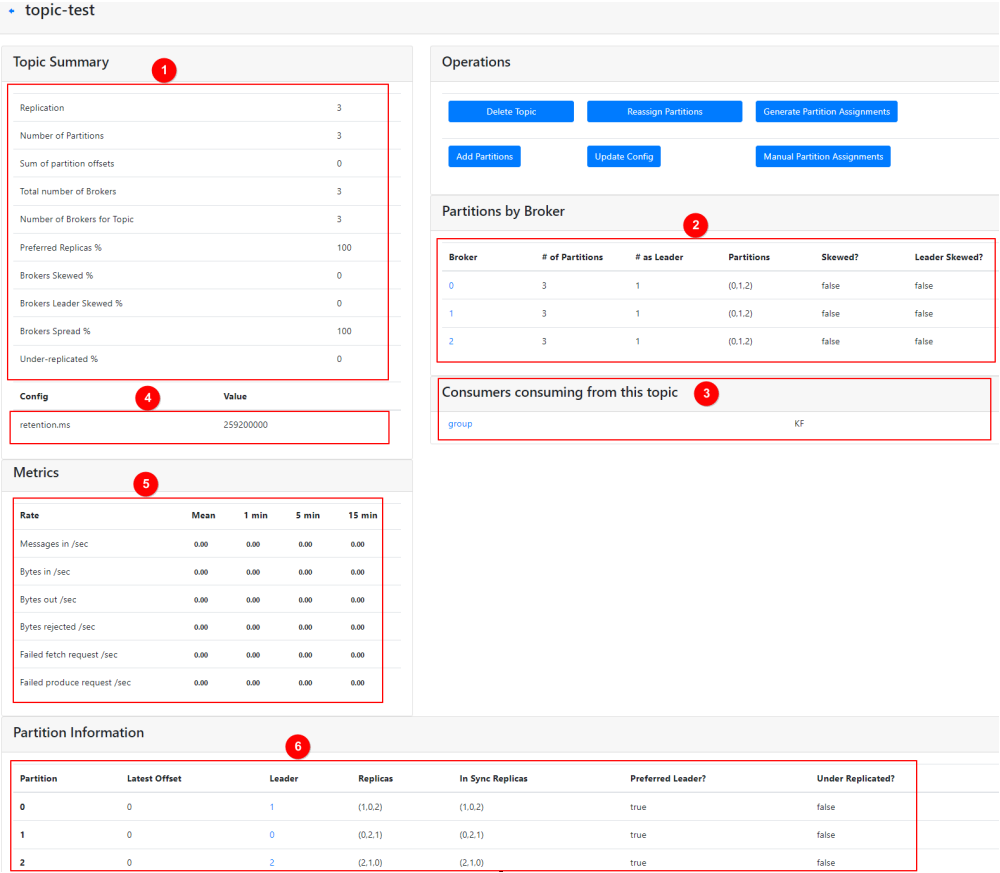
- 查看实例的Topic  
在导航栏选择**Topic**，并在下拉列表中选择**List**。页面如图9-7所示，展示了队列列表以及分区数等。  
列表中以“\_\_”开头的队列为内部队列，严禁操作，否则可能导致业务问题。

图 9-7 查看实例的 Topic



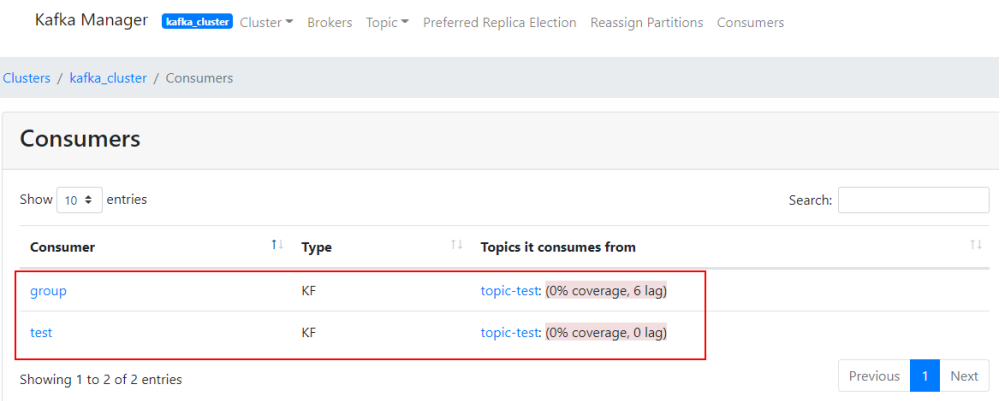
- 队列详情页  
单击具体的Topic名称，进入如图9-8所示页面。
  - 图中①区域表示队列基本信息，包括副本数(Replication)，分区数(Number of Partitions)，消息数(Sum of partition offsets)等。
  - 图中②区域表示代理与队列分区的对应关系。
  - 图中③区域表示该队列的消费组列表。单击消费组名称可进入该消费组的详情页。
  - 图中④区域表示队列的配置信息。详情参考[Kafka队列官方配置文档](#)。
  - 图中⑤区域表示队列监控数据统计。
  - 图中⑥区域表示队列分区信息，包括分区消息数(Latest Offset)，分区 leader(Leader)，副本列表(Replicas)，同步副本列表(In Sync Replicas)。

图 9-8 队列详情页



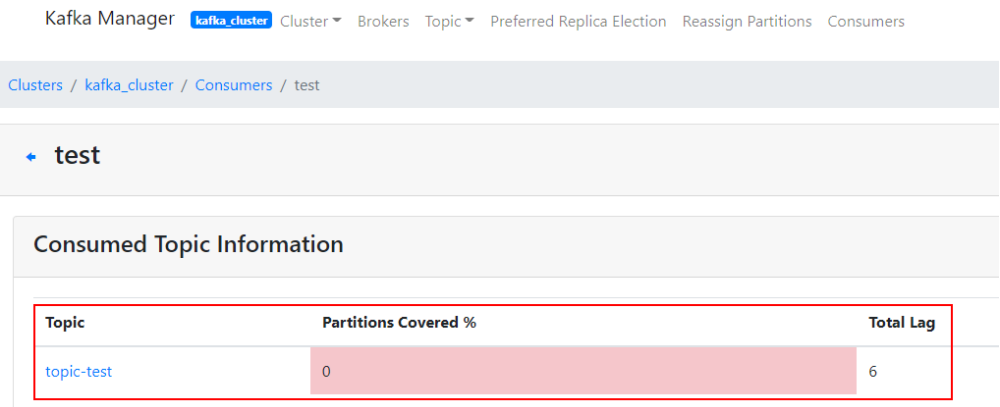
- 查看消费组列表  
导航栏中单击**Consumers**，即可查看当前集群中的消费组列表。此页面只显示14天内有消费记录的消费组。

图 9-9 集群的消费组列表



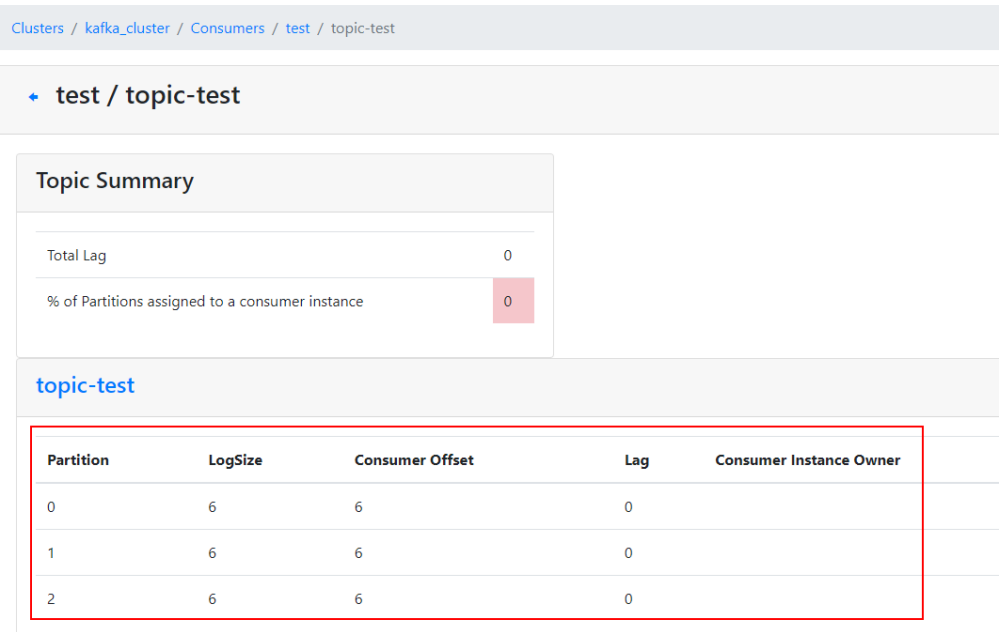
- 查看消费组详情页  
单击消费组名称可进入消费组详情页面，展示消费组消费的所有队列列表以及每个队列的可消费数（Total Lag）。

图 9-10 消费组详情页面



- 查看消费组队列详情页  
单击队列名称，即可进入详情页，查看消费组消费在队列中每个分区的消费状态。包括分区编号（Partition）、分区消息数（LogSize）、分区消费进度（Consumer Offset）、分区剩余可消费数（Lag）和最近消费该分区的消费者（Consumer Instance Owner）。

图 9-11 消费组队列详情页



### 9.12.2 重置 Kafka Manager 密码

如果您忘记了创建实例时设置的Kafka Manager密码，通过重置Manager密码功能，重新设置一个新的密码，可使用新密码登录Kafka Manager。

2023年5月17日及以后创建的实例不提供重置Kafka Manager密码功能。

#### 前提条件

已创建Kafka实例，且实例处于“运行中”。

## 重置 Kafka Manager 密码

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，重置Kafka Manager密码。

- 在待重置Manager密码的Kafka实例所在行，单击“更多 > 重置Manager密码”。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 重置Manager密码”。
- 单击Kafka实例名称，进入实例详情页面。在“概览”页面，在“连接信息 > Manager用户名”后，单击“重置Manager密码”。

**步骤4** 输入“新密码”和“确认密码”，单击“确定”，完成密码重置。

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

### 说明

只有所有代理都重置密码成功，系统才会提示重置密码成功，否则会提示重置失败。

----结束

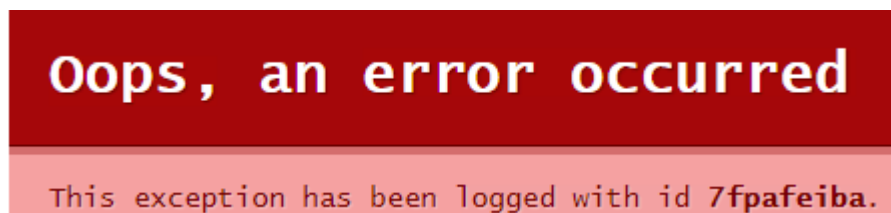
## 相关文档

如果您希望通过API重置Kafka Manager密码，请参考[重置Manager密码](#)。

### 9.12.3 重启 Kafka Manager

当Kafka Manager无法登录或者无法使用时，例如下图中的报错，可以通过重启Kafka Manager，使Kafka Manager恢复正常。

图 9-12 报错信息



2023年5月17日及以后创建的实例不提供重启Kafka Manager功能。

## 约束与限制

重启Kafka Manager不会影响业务。

## 重启 Kafka Manager

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，重启Kafka Manager。

- 在待重启Manager的Kafka实例所在行，单击“更多 > 重启Manager”。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 重启Manager”。

**步骤4** 单击“是”。

您可以在实例的“实例管理 > 后台任务管理”页面，查看当前任务的操作进度。任务状态为“成功”，表示重启成功。

----结束

## 相关文档

如果您希望通过API重启Kafka Manager，请参考[重启Manager](#)。

## 9.12.4 关闭 Kafka Manager

Kafka Manager会占用额外的内存和CPU资源，关闭Kafka Manager可以减少这部分资源的占用。本章节指导您在控制台关闭Kafka Manager。

### 约束与限制

- Kafka Manager关闭后，无法再次开启。
- 关闭Kafka Manager，不会重启实例。

### 关闭 Kafka Manager

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击待关闭Kafka Manager的Kafka实例名称，进入实例详情页面。

**步骤4** 在“概览”页面的“连接信息 > Kafka Manager”后，单击 ，关闭Kafka Manager。

关闭Kafka Manager后，控制台不显示Kafka Manager连接地址，不支持重置Kafka Manager密码和重启Kafka Manager功能。

----结束

## 相关文档

如果您希望通过API关闭Kafka Manager，请参考[关闭Kafka Manager](#)。

# 10 变更实例规格

## 10.1 变更 Kafka 集群实例规格

Kafka实例创建成功后，您可以根据业务需要，扩容或者缩容实例规格，Kafka实例支持的变更配置如表10-1所示。

表 10-1 变更配置列表

| 实例规格类型 | 变更配置类型 | 是否支持扩容 | 是否支持缩容 |
|--------|--------|--------|--------|
| 新规格实例  | 代理数量   | √      | ×      |
|        | 存储空间   | √      | ×      |
|        | 代理规格   | √      | √      |
| 老规格实例  | 基准带宽   | √      | ×      |
|        | 存储空间   | √      | ×      |
|        | 代理规格   | ×      | ×      |

### 怎样判断老规格实例和新规格实例？

- 老规格实例：在实例列表中，“规格”显示为带宽（如100MB/s）的实例。
- 新规格实例：在实例列表中，“规格”显示为ECS规格和代理个数（如 kafka.2u4g.cluster\*3 broker）的实例。

图 10-1 实例列表

| 名称                                           | 状态  | 版本    | 规格                            | 已用/可用存储空间 (GB) |
|----------------------------------------------|-----|-------|-------------------------------|----------------|
| kafka-test<br>a0af5cb3-c7fd-4f2d-ac1f-fe9... | 运行中 | 2.7   | kafka.2u4g.cluster * 3 broker | 0/300          |
| kafka-01<br>252b248f-501e-41ed-9832-48       | 运行中 | 2.3.0 | 100 MB/s                      | 0/186          |

约束与限制

- 单机实例不支持变更实例规格。
- 变更实例规格后，配置费用将发生变化。

变更实例规格的影响

表 10-2 变更实例规格的影响

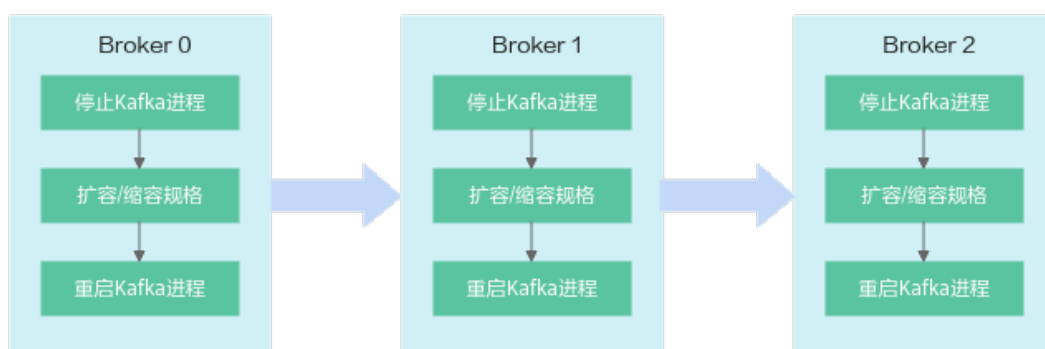
| 变更配置类型    | 影响                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 基准带宽/代理数量 | <ul style="list-style-type: none"><li>• 扩容基准带宽/代理数量不会影响原来的代理，<b>业务也不受影响</b>。</li><li>• 扩容基准带宽/代理数量时，系统会根据当前磁盘大小进行相应比例的存储空间扩容。例如扩容前实例的代理数为3，每个代理的磁盘大小为200GB，扩容后实例的代理数为10，此时代理的磁盘大小依旧为200GB，但是总磁盘大小为2000GB。</li><li>• <b>新创建的Topic才会分布在新代理上</b>，原有Topic还分布在原有代理上，造成分区分布不均匀。通过<a href="#">修改Kafka分区平衡</a>，实现将原有Topic分区的副本迁移到新代理上。</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 存储空间      | <ul style="list-style-type: none"><li>• 扩容存储空间有次数限制，只能扩容<b>20次</b>。</li><li>• <b>扩容存储空间不会影响业务</b>。</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 代理规格      | <ul style="list-style-type: none"><li>• 若Topic为<b>单副本</b>，扩容/缩容期间无法对该Topic生产消息或消费消息，会造成<b>业务中断</b>。</li><li>• 若Topic为<b>多副本</b>，扩容/缩容代理规格<b>不会造成服务中断</b>，但会逐个节点重启，负载会转移到剩余节点上，建议您在业务低峰期扩容/缩容。</li><li>• 扩容/缩容代理规格的过程中，节点滚动重启造成分区Leader切换，会发生<b>秒级连接闪断</b>，在用户网络环境稳定的前提下，Leader切换时长一般为<b>1分钟以内</b>。多副本的Topic需要在生产客户端配置<b>重试机制</b>，方法如下：<ul style="list-style-type: none"><li>- 生产客户端为<b>Kafka开源客户端</b>时，检查是否配置retries和retry.backoff.ms参数。建议参数值分别配置为：retries=10，retry.backoff.ms=1000。</li><li>- 生产客户端为<b>Flink客户端</b>时，检查是否配置重启策略，配置重启策略可以参考如下代码。<pre>StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvironment(); env.setRestartStrategy(RestartStrategies.fixedDelayRestart(10, Time.seconds(20)));</pre></li></ul></li><li>• 若实例已创建的分区数总和大于待缩容规格的实例分区数上限，此时无法缩容。不同规格配置的实例分区数上限不同，具体请参见<a href="#">产品规格</a>。<br/>例如：kafka.4u8g.cluster*3的实例，已创建800个分区，您想把此实例的规格缩容为kafka.2u4g.cluster*3，kafka.2u4g.cluster*3规格的实例分区数上限为750，此时无法缩容。</li><li>• 变更代理规格的时长和代理数量有关，单个代理的变更时长一般在<b>5~10分钟</b>，代理数量越多，规格变更时间越长。</li></ul> |

## 扩容/缩容代理规格的过程

在扩容/缩容代理规格的过程中，代理采用滚动重启的方式进行实例变更，具体过程如下（以3个代理为例介绍）：

1. 停止Broker 0的Kafka进程。
2. 扩容/缩容Broker 0的规格。
3. 重启Broker 0的Kafka进程。
4. 重复1~3，扩容/缩容Broker 1的规格。
5. 重复1~3，扩容/缩容Broker 2的规格。

图 10-2 扩容/缩容代理规格过程



## 变更 Kafka 集群实例规格

以下介绍变更Kafka集群实例规格的具体步骤。

代理数量/基准带宽、存储空间、代理规格这三类规格不能同时变更，每次仅支持变更一种类型。

### 扩容代理数量/基准带宽

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在Kafka实例所在行，单击“变更规格”，进入“分布式消息服务Kafka版变更规格”页面。

**步骤4** 根据实际情况选择扩容代理数量或基准带宽。

- **扩容基准带宽（[如何判断老规格实例](#)）**

- a. 在“规格”中，选择扩容后的带宽，单击“下一步”。
- b. 确认扩容信息无误后，单击“提交”。
- c. 在实例列表页面，查看变更是否成功？
  - 实例状态从“变更中”变为“运行中”时，表示变更成功，在“规格”中查看扩容后的带宽大小。

图 10-3 查看扩容后的带宽

| <input type="checkbox"/> | 名称                                                | 状态  | 版本    | 规格       |
|--------------------------|---------------------------------------------------|-----|-------|----------|
| <input type="checkbox"/> | kafka-1674331489<br>5b0ac100-aa24-4461-8950-dc... | 运行中 | 2.3.0 | 300 MB/s |

- 实例状态从“变更中”变为“变更失败”时，表示变更失败，把鼠标放在“变更失败”上，显示变更失败的原因。  
实例状态为“变更失败”时，无法对实例进行重启、修改和删除等操作。等一段时间实例状态会自动从“变更失败”变为“运行中”，此时可以继续对实例进行操作。如果未变为“运行中”，请联系客服处理。
  - d. 扩容基准带宽后，为了提高可靠性，需要在客户端连接配置中增加新扩容代理的IP地址。
- 扩容代理数量（[如何判断新规格实例](#)）

图 10-4 扩容代理数量

变更类型

存储空间代理数量代理规格

代理数量不支持扩容。代理扩容后可以分区平衡 [查看详情](#)

代理数量

-4+

当前实例磁盘总带宽规格：460 MB/s（当前实例磁盘总带宽规格 = 单个代理磁盘规格 \* 代理数量）  
单个代理磁盘带宽：115 MB/s

子网

subnet- (192.168.0.0/24) 可用 IP 248 个

内网IP地址

自动分配手动选择

变更后规格

总存储空间 400 GB | 单个代理存储空间 高I/O 100 GB | 代理数量 4

- a. 在“变更类型”中，选择“代理数量”。
- b. 在“代理数量”中，选择扩容后的代理数量。不同实例规格对应的代理数量范围不同，具体请参考[Kafka集群实例规格](#)。
- c. 如果已开启公网访问，在“弹性IP地址”中，选择新扩容代理的公网IP地址。
- d. “子网”保持默认设置。
- e. 在“内网IP地址”中，选择“自动分配”或者“手动选择”。
  - 自动分配：系统自动分配子网中可用的IP地址。
  - 手动选择：在下拉框中为新扩容代理勾选指定的IP地址。如果指定的IP地址数量少于代理数量时，系统会自动为剩余代理分配IP地址。
- f. 单击“下一步”，进入变更信息确认页面。
- g. 确认扩容信息无误后，单击“提交”。
- h. 查看变更进度和预计剩余时间。
- i. 在实例列表页面，单击实例名称，进入实例详情页。
- ii. 在左侧导航栏单击“实例管理 > 后台任务管理”，进入“即时任务”页签。

- iii. 单击“任务名”为“变更规格”的任务，弹出“变更规格任务详情”对话框。
- iv. 在顶部区域，查看变更进度和预计剩余时间。在“任务执行步骤”中，查看执行步骤的名称和开始执行时间，以及结束时间。
- i. 查看变更是否成功。

■ 变更规格任务的状态为“成功”时，表示变更成功。在实例列表页面的“规格”中查看扩容后的代理数量。

■ 变更规格任务的状态为“失败”时，表示变更失败。把鼠标放在“失败”上，显示变更失败的原因。也可以在“任务执行步骤”中查看变更失败的原因。

变更失败后，实例状态会变为“变更失败”，此时无法对实例进行重启、修改和删除等操作。等一段时间实例状态会自动从“变更失败”变为“运行中”，此时可以继续对实例进行操作。如果未变为“运行中”，请联系客服处理。
- j. 扩容代理数量后，为了提高可靠性，需要在客户端连接配置中增加新扩容代理的IP地址。

----结束

扩容存储空间

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 在Kafka实例所在行，单击“变更规格”，进入“分布式消息服务Kafka版变更规格”页面。
- 步骤4 根据实际情况选择扩容存储空间。

● 扩容存储空间（[如何判断老规格实例](#)）

a. 在“存储空间”中，选择扩容后的存储空间大小，单击“下一步”。

b. 确认扩容信息无误后，单击“提交”。

c. 在实例列表页面，查看变更是否成功？

■ 实例状态从“变更中”变为“运行中”时，表示变更成功，在“已用/可用存储空间”中查看扩容后的存储空间大小。

图 10-5 查看扩容后的存储空间

| 名称                                                               | 状态             | 版本    | 规格       | 已用/可用存储空间...          |
|------------------------------------------------------------------|----------------|-------|----------|-----------------------|
| <div>kafka-1674331489</div> <div>5b0ac100-aa24-4461-895...</div> | <div>运行中</div> | 2.3.0 | 100 MB/s | <div>0/558 (0%)</div> |

- 实例状态从“变更中”变为“变更失败”时，表示变更失败，把鼠标放在“变更失败”上，显示变更失败的原因。
- 实例状态为“变更失败”时，无法对实例进行重启、修改和删除等操作。等一段时间实例状态会自动从“变更失败”变为“运行中”，此时可以继续对实例进行操作。如果未变为“运行中”，请联系客服处理。

- 扩容存储空间（[如何判断新规格实例](#)）

图 10-6 扩容存储空间

变更类型 **存储空间** 代理数量 代理规格

存储空间不支持缩容。

单个代理存储空间 高I/O  GB

当前实例磁盘总带宽规格：390 MB/s（当前实例磁盘总带宽规格 = 单个代理磁盘规格 \* 代理数量）

单个代理磁盘带宽：130 MB/s

变更后规格 总存储空间 600 GB | 单个代理存储空间 高I/O 200 GB | 代理数量 3

- 在“变更类型”中，选择“存储空间”。
- 在“单个代理存储空间”中，选择扩容后的单个代理的存储空间大小，不同实例规格对应的存储空间范围不同，具体请参考[Kafka集群实例规格](#)，单击“下一步”。
- 确认扩容信息无误后，单击“提交”。
- 查看变更进度和预计剩余时间。
  - 在实例列表页面，单击实例名称，进入实例详情页。
  - 在左侧导航栏单击“实例管理 > 后台任务管理”，进入“即时任务”页签。
  - 单击“任务名”为“变更规格”的任务，弹出“变更规格任务详情”对话框。
  - 在顶部区域，查看变更进度和预计剩余时间。在“任务执行步骤”中，查看执行步骤的名称和开始执行时间，以及结束时间。
- 查看变更是否成功。
  - 变更规格任务的状态为“成功”时，表示变更成功。在实例列表页面的“可用存储空间”中查看扩容后的存储空间大小（即扩容后的单个代理的存储空间\*代理数量）。
  - 变更规格任务的状态为“失败”时，表示变更失败。把鼠标放在“失败”上，显示变更失败的原因。也可以在“任务执行步骤”中查看变更失败的原因。

变更失败后，实例状态会变为“变更失败”，此时无法对实例进行重启、修改和删除等操作。等一段时间实例状态会自动从“变更失败”变为“运行中”，此时可以继续对实例进行操作。如果未变为“运行中”，请联系客服处理。

----结束

## 扩容/缩容代理规格

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

- 步骤3 在Kafka实例所在行，单击“变更规格”，进入“分布式消息服务Kafka版变更规格”页面。
- 步骤4 根据实际情况选择扩容/缩容代理规格。

图 10-7 扩容/缩容代理规格

变更类型

存储空间

代理数量

代理规格

变更代理规格的过程中，节点会重启，可能造成闪断；若topic为单副本，则会导致期间无法对该topic生产消息。

| 规格名称                                                | 单个代理TPS | 单个代理最大分区数 | 单个代理消费组数 | 单个代理流量规格 (MB/s) |
|-----------------------------------------------------|---------|-----------|----------|-----------------|
| <input checked="" type="radio"/> kafka.2u4g.cluster | 30,000  | 250       | 20       | 100             |
| <input type="radio"/> kafka.4u8g.cluster            | 100,000 | 500       | 100      | 200             |
| <input type="radio"/> kafka.8u16g.cluster           | 150,000 | 1,000     | 150      | 375             |
| <input type="radio"/> kafka.12u24g.cluster          | 200,000 | 1,500     | 200      | 625             |
| <input type="radio"/> kafka.16u32g.cluster          | 250,000 | 2,000     | 200      | 750             |

当前实例网络总流量规格：400 MB/s（当前实例网络总流量规格 = 单个代理流量规格 \* 代理数量）

当前实例业务流量规格：

+ 建议读流量：66.7 MB/s

+ 建议写流量：66.7 MB/s

实例网络总流量规格和业务流量规格参考说明请参考 [实例规格和网络带宽说明](#)。

由于一般业务系统流量存在波动，建议您预留30%冗余流量，防止出现流量波动影响您的业务。

变更后规格

kafka.2u4g.cluster

1. 在“变更类型”中，选择“代理规格”。
2. 在“代理规格”中，选择扩容/缩容后的代理规格。
3. 在“风险检查”区域，查看扩容/缩容前的检查项是否正常。  
如果检查项有异常，请先根据提示信息修改，然后单击“重新检查”。如果确认异常的检查项无需修改，请勾选“我已知晓风险”。
4. 单击“下一步”，确认扩容/缩容信息无误后，单击“提交”。
5. 查看扩容进度和预计剩余时间。

a. 在实例列表页面，单击实例名称，进入实例详情页。

b. 在左侧导航栏单击“实例管理 > 后台任务管理”，进入“即时任务”页签。

c. 单击“任务名”为“变更规格”的任务，弹出“变更规格任务详情”对话框。

d. 在顶部区域，查看变更进度和预计剩余时间。在“任务执行步骤”中，查看执行步骤的名称和开始执行时间，以及结束时间。
6. 查看变更是否成功。

- 变更规格任务的状态为“成功”时，表示变更成功。在实例列表页面的“规格”中查看扩容/缩容后的代理规格。

- 变更规格任务的状态为“失败”时，表示变更失败。把鼠标放在“失败”上，显示变更失败的原因。如果是扩容失败，也可以在“任务执行步骤”中查看变更失败的原因。

变更失败后，实例状态会变为“变更失败”，此时无法对实例进行重启、修改和删除等操作。等一段时间实例状态会自动从“变更失败”变为“运行中”，此时可以继续对实例进行操作。如果未变为“运行中”，请联系客服处理。

----结束

相关文档

如果您希望通过API变更Kafka集群实例规格，请参考[实例扩容](#)。

## 10.2 自动扩容 Kafka 实例存储空间

Kafka实例的磁盘使用量达到容量阈值时会触发对应的阈值处理策略，可能会导致消息被删除或无法继续生产消息。为了避免以上情况，您可以开启自动磁盘扩容功能。

### 约束与限制

- 老实例不支持自动磁盘扩容功能，具体以控制台显示为准。
- 单机实例不支持自动磁盘扩容。
- 如果短时间内磁盘使用量迅速增加，无法保证在磁盘使用量到达容量阈值前完成自动磁盘扩容。
- 自动扩容存储空间不会影响业务。

### 前提条件

扩容Kafka实例存储空间会产生相应的费用，确保账户有充足的余额，避免因余额不足导致扩容失败。Kafka实例存储空间的具体价格请参考[价格详情](#)。

### 为 Kafka 授权

开启自动扩容存储空间前，需要先获取“DMS BSSAccess”权限，否则会导致自动扩容失败。

**步骤1** 使用主账号或者“admin”用户组中的子账号登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择区域。

**步骤3** 在左侧导航栏选择“服务授权”，进入“服务授权”页面。

**步骤4** 勾选“访问费用中心服务”，单击“更新授权”。

**步骤5** 授权成功后，会在IAM中创建名为“dms\_admin\_trust”的委托。

“dms\_admin\_trust”创建成功后，请勿修改或者删除。

----结束

### 开启自动扩容存储空间

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入Kafka实例的概览页面。

**步骤4** 在“实例信息”区域的“自动磁盘扩容”后，单击，弹出“自动磁盘扩容”对话框。

**步骤5** 单击，并设置如[表10-3](#)所示参数。

表 10-3 自动磁盘扩容参数说明

| 参数   | 说明                                                                                                                                                                        |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 扩容阈值 | 设置触发磁盘自动扩容的阈值，取值范围：20%~80%。<br>当Kafka集群实例中任意一个节点的磁盘使用量达到此阈值时，会自动触发磁盘扩容。                                                                                                   |
| 扩容步长 | 设置待扩容的存储空间占实例总存储空间的比例，取值范围：10%-100%。 <b>最小扩容步长为100GB，如果待扩容的存储空间不是100的倍数时，补足100。</b><br>例如：实例总存储空间为300GB，扩容步长为10%，待扩容的存储空间=300*10%=30，由于30不是100的倍数，补足100，所以待扩容的存储空间为100GB。 |
| 扩容上限 | 设置自动扩容存储空间的上限，实例总存储空间到达此上限后，实例不会再自动扩容存储空间。                                                                                                                                |

**步骤6** 单击“确定”，完成自动磁盘扩容的设置。

设置成功后，在Kafka实例概览页面的“实例信息”区域，查看“自动磁盘扩容”变为“开启”。

----结束

关闭自动扩容存储空间

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例的名称，进入Kafka实例的概览页面。

**步骤4** 在“实例信息”区域的“自动磁盘扩容”后，单击，弹出“自动磁盘扩容”对话框。

**步骤5** 在“自动磁盘扩容”所在行，单击，然后单击“确定”。

设置成功后，在Kafka实例概览页面的“实例信息”区域，查看“自动磁盘扩容”变为“关闭”。

----结束

# 11 迁移数据

## 11.1 Kafka 数据迁移概述

Kafka迁移是指将生产与消费消息的客户端切换为连接新Kafka，部分还涉及将持久化的消息文件迁移到新Kafka。主要涉及到以下2类场景：

- 业务上云且不希望业务有中断。  
在上云过程中，连续性要求高的业务，需要平滑迁移，不能有长时间的中断。
- 在云上变更业务部署。  
单AZ部署的Kafka实例，不具备AZ之间的容灾能力。用户对可靠性要求提升后，需要迁移到多AZ部署的实例上。

### 迁移方案概述

表 11-1 迁移方案概述

| 迁移方案             | 迁移工具 | 优点                                                                                                                    | 缺点                                                                                                                          |
|------------------|------|-----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| 先迁生产，再迁消费（不迁移数据） | -    | <ul style="list-style-type: none"><li>业界通用的迁移方案，操作步骤简单，无需安装其他插件。</li><li>迁移过程由业务侧自主控制。</li><li>可确保消息消费的顺序性。</li></ul> | <ul style="list-style-type: none"><li>切换过程存在延时，消费端需要先消费完原Kafka消息，然后才能消费新Kafka消息。</li><li>迁移完消费业务后，新Kafka可能存在消息积压。</li></ul> |
| 同时消费，后迁生产（不迁移数据） | -    | <ul style="list-style-type: none"><li>操作步骤简单，无需安装其他插件。</li><li>同时消费原Kafka和新Kafka消息，确保消息能及时消费，无消息积压。</li></ul>         | 在迁移生产的开始阶段，同时消费原Kafka与新Kafka，会导致部分消息之间的生产顺序无法保证，存在消息乱序的问题。                                                                  |

| 迁移方案                             | 迁移工具          | 优点                               | 缺点                                  |
|----------------------------------|---------------|----------------------------------|-------------------------------------|
| 使用MirrorMaker先迁移数据，然后迁消费，最后迁生产   | Mirror Maker  | 新Kafka同步原Kafka的全量历史数据和实时生产的增量数据。 | 消费端迁移到新Kafka后，会重复消费历史消息，消费端需支持消息幂等。 |
| 使用Smart Connect先迁移数据，然后迁消费，最后迁生产 | Smart Connect | 新Kafka同步原Kafka的全量历史数据和实时生产的增量数据。 | 消费端迁移到新Kafka后，会重复消费历史消息，消费端需支持消息幂等。 |

约束与限制

- 使用Smart Connect迁移业务，会对源端Kafka进行消费，对目标端Kafka进行生产，会占用源端和目标端Kafka的带宽。
- 出于性能考虑，Smart Connect实时同步源端和目标端的数据，但是消费进度是通过批处理同步的，可能会导致源端和目标端每个分区的消费进度存在0-100之间的差异。

迁移准备

1. 配置网络环境。
- Kafka实例分内网地址以及公网地址两种网络连接方式。如果使用公网地址，则消息生成与消费客户端需要有公网访问权限，并配置如下安全组。

表 11-2 安全组规则

| 方向  | 协议  | 端口   | 源地址                 | 说明                 |
|-----|-----|------|---------------------|--------------------|
| 入方向 | TCP | 9094 | Kafka客户端所在的IP地址或地址组 | 通过公网访问Kafka（明文接入）。 |
| 入方向 | TCP | 9095 | Kafka客户端所在的IP地址或地址组 | 通过公网访问Kafka（密文接入）。 |

2. 创建目标Kafka实例。
- 目标Kafka的规格不能低于原业务使用的Kafka规格。具体请参考[购买Kafka实例](#)。
3. 在目标Kafka实例中创建Topic。
- 在目标Kafka实例上创建与原Kafka实例相同配置的Topic，包括Topic名称、副本数、分区数、消息老化时间，以及是否同步复制和落盘等。具体请参考[创建Kafka Topic](#)。

## 迁移方案一：先迁生产，再迁消费（不迁移数据）

先将生产消息的业务迁移到新Kafka，原Kafka不会有新消息生产。待原Kafka的消息全部消费完后，再将消费消息业务迁移到新Kafka，开始消费新Kafka的消息。

本方案为业界通用的迁移方案，操作步骤简单，迁移过程由业务侧自主控制，整个过程中消息不会存在乱序问题，**适用于对消息顺序有要求的场景**。但是该方案中需要等待消费业务执行完毕，存在一个时间差的问题，部分数据可能存在较大的端到端时延。

- 步骤1** 将生产客户端的Kafka连接地址修改为新Kafka实例的连接地址。
- 步骤2** 重启生产业务，使得生产者将新的消息发送到新Kafka实例中。
- 步骤3** 观察各消费组在原Kafka的消费进度，直到原Kafka中数据都已经被消费完毕。
- 步骤4** 将消费客户端的Kafka连接地址修改为新Kafka实例的连接地址。
- 步骤5** 重启消费业务，使得消费者从新Kafka实例中消费消息。
- 步骤6** 观察消费者是否能正常从新Kafka实例中获取数据。
- 步骤7** 迁移结束。

----结束

## 迁移方案二：同时消费，后迁生产（不迁移数据）

消费业务启用多个消费客户端，分别向原Kafka和新Kafka消费消息，然后将生产业务切到新Kafka实例，这样能确保所有消息都被及时消费。

本方案中消费业务会在一段时间内同时消费原Kafka和新Kafka消息。由于在迁移生产业务之前，已经有消费业务运行在新Kafka实例上，因此不会存在端到端时延的问题。但在迁移生产的开始阶段，同时消费原Kafka与新Kafka，会导致部分消息之间的生产顺序无法保证，存在消息乱序的问题。此场景**适用于对端到端时延有要求，但对消息顺序不敏感的业务**。

- 步骤1** 启动新的消费客户端，配置Kafka连接地址为新Kafka实例的连接地址，消费新Kafka实例中的数据。  
  
原有消费客户端需继续运行，消费业务同时消费原Kafka与新Kafka实例的消息。
- 步骤2** 修改生产客户端，Kafka连接地址改为新Kafka实例的连接地址。
- 步骤3** 重启生产客户端，将生产业务迁移到新Kafka实例中。
- 步骤4** 生产业务迁移后，观察连接新Kafka实例的消费业务是否正常。
- 步骤5** 等待原Kafka中数据消费完毕，关闭原有消费业务客户端。
- 步骤6** 迁移结束。

----结束

## 迁移方案三：使用 MirrorMaker 先迁移数据，然后迁消费，最后迁生产

首先通过MirrorMaker同步两个Kafka的消息，其次将消费端迁移到新Kafka，最后将生产端迁移到新Kafka。

本方案依赖于MirrorMaker，MirrorMaker实时同步原Kafka和新Kafka的数据。待数据同步完成，先将消费端迁移到新Kafka，然后将生产端迁移到新Kafka。此场景适用于生产端不可停止，端到端有时延要求，但是可以兼容少量重复消费的业务。

- 步骤1** 使用MirrorMaker同步两个Kafka的消息。具体步骤请参见[使用MirrorMaker跨集群同步数据](#)。
- 步骤2** 将消费客户端的Kafka连接地址修改为新Kafka实例的连接地址。
- 步骤3** 重启消费业务，使得消费者从新Kafka实例中消费消息。
- 步骤4** 观察消费者是否能正常从新Kafka实例中获取数据。
- 步骤5** 修改生产客户端，Kafka连接地址改为新Kafka实例的连接地址。
- 步骤6** 重启生产客户端，将生产业务迁移到新Kafka实例中。
- 步骤7** 生产业务迁移后，观察连接新Kafka实例的消费业务是否正常。
- 步骤8** 迁移结束。

----结束

## 迁移方案四：使用 Smart Connect 先迁移数据，然后迁消费，最后迁生产

首先通过Smart Connect同步两个Kafka的消息，其次将消费端迁移到新Kafka，最后将生产端迁移到新Kafka。

本方案依赖于Smart Connect，Smart Connect实时同步源端和目标端的数据，但是消费进度是通过批处理同步的，可能会导致源端和目标端每个分区的消费进度存在0-100之间的差异，存在少量重复消费问题。此场景适用于生产端不可停止，端到端有时延要求，但是可以兼容少量重复消费的业务。

- 步骤1** 创建Kafka数据复制的Smart Connect任务，用于同步两个Kafka的消息。具体步骤请参见[配置Kafka间的数据复制](#)。
- 步骤2** 在Kafka控制台的“实例管理 > 消息查询”页面，查看两个Kafka的最新消息是否一致，确认两个Kafka的同步进度是否一致。具体步骤请参见[查看Kafka消息](#)。
  - 是，执行[步骤3](#)。
  - 否，在监控页面查看两个Kafka的“Kafka每分钟同步数据量”是否正常，如果正常，先等待两个Kafka的同步进度一致，然后执行[步骤3](#)。
- 步骤3** 将消费客户端的Kafka连接地址修改为新Kafka实例的连接地址。
- 步骤4** 重启消费业务，使得消费者从新Kafka实例中消费消息。
- 步骤5** 观察消费者是否能正常从新Kafka实例中获取数据。
- 步骤6** 修改生产客户端，Kafka连接地址改为新Kafka实例的连接地址。
- 步骤7** 重启生产客户端，将生产业务迁移到新Kafka实例中。
- 步骤8** 生产业务迁移后，观察连接新Kafka实例的消费业务是否正常。
- 步骤9** 迁移结束。

----结束

## 常见问题：如何将持久化数据也一起迁移

如果需要将原Kafka的已消费数据也迁移到Kafka实例，可以使用Smart Connect工具或者开源工具[MirrorMaker](#)，模拟成原Kafka的消费客户端，以及新Kafka实例的生产客户端，将Kafka所有消息数据迁移到新的Kafka实例，具体步骤请参考[配置Kafka间的数据复制](#)或者[使用MirrorMaker跨集群同步数据](#)。

## 11.2 使用 MirrorMaker 跨集群同步数据

### 方案概述

#### 应用场景

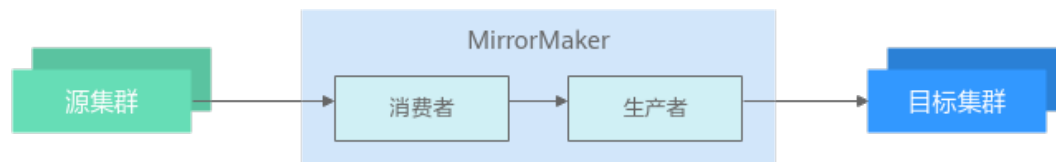
在以下场景，使用MirrorMaker进行不同集群间的数据同步，可以确保Kafka集群的可用性和可靠性。

- 备份和容灾：企业存在多个数据中心，为了防止其中一个数据中心出现问题，导致业务不可用，会将集群数据同步备份在多个不同的数据中心。
- 集群迁移：当今很多企业将业务迁移上云，迁移过程中需要确保线下集群和云上集群的数据同步，保证业务的连续性。

#### 方案架构

使用MirrorMaker可以实现将源集群中的数据镜像复制到目标集群中。其原理如[图 11-1](#)所示，MirrorMaker本质上也是生产消费消息，首先从源集群中消费数据，然后将消费的数据生产到目标集群。如果您需要了解更多关于MirrorMaker的信息，请参见[Mirroring data between clusters](#)。

图 11-1 MirrorMaker



### 约束与限制

- 源集群中节点的IP地址和端口号不能和目标集群中节点的IP地址和端口号相同，否则会导致数据在Topic内无限循环复制。
- 使用MirrorMaker同步数据，至少需要有两个或以上集群，不可在单个集群内部使用MirrorMaker，否则会导致数据在Topic内无限循环复制。

### 实施步骤

**步骤1** 购买一台弹性云服务器，确保弹性云服务器与源集群、目标集群网络互通。具体购买操作，请参考[购买弹性云服务器](#)。

**步骤2** 登录弹性云服务器，安装Java JDK，并配置JAVA\_HOME与PATH环境变量，使用执行用户在用户家目录下修改“.bash\_profile”，添加如下行。其中“/opt/java/jdk1.8.0\_151”为JDK的安装路径，请根据实际情况修改。

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

执行`source .bash_profile`命令使修改生效。

### 说明

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

**步骤3** 下载Kafka 3.3.1版本的二进制软件包。

```
wget https://archive.apache.org/dist/kafka/3.3.1/kafka_2.12-3.3.1.tgz
```

**步骤4** 解压二进制软件包。

```
tar -zxvf kafka_2.12-3.3.1.tgz
```

**步骤5** 进入二进制软件包目录，修改“config”目录下的“connect-mirror-maker.properties”的配置文件，在配置文件中指定源集群和目标集群的IP地址和端口以及其他配置。

```
指定两个集群
clusters = A, B
A.bootstrap.servers = A_host1:A_port, A_host2:A_port, A_host3:A_port
B.bootstrap.servers = B_host1:B_port, B_host2:B_port, B_host3:B_port

指定数据同步方向，可以单向同步也可互相同步
A->B.enabled = true

指定同步的Topic，支持正则匹配，默认复制全部Topic，如："foo-.*"
A->B.topics = .*

取消以下两个配置的注释表示A、B两个集群互相复制同步
#B->A.enabled = true
#B->A.topics = .*

设置副本个数，如果需要同步多个Topic且副本数各不相同，建议先创建同名同副本数的Topic再启动
MirrorMaker
replication.factor=3

设置消费进度同步方向，可以单向同步也可互相同步
A->B.sync.group.offsets.enabled=true

Internal Topic Settings
The replication factor for mm2 internal topics "heartbeats", "B.checkpoints.internal" and
"mm2-offset-syncs.B.internal"
测试环境可以为1，生产环境建议以下配置大于1，比如设为3
checkpoints.topic.replication.factor=3
heartbeats.topic.replication.factor=3
offset-syncs.topic.replication.factor=3

The replication factor for connect internal topics "mm2-configs.B.internal", "mm2-offsets.B.internal" and
"mm2-status.B.internal"
测试环境可以为1，生产环境建议以下配置大于1，比如设为3
offset.storage.replication.factor=3
status.storage.replication.factor=3
config.storage.replication.factor=3

customize as needed
replication.policy.separator = _
sync.topic.acls.enabled = false
emit.heartbeats.interval.seconds = 5
设置目标集群中的Topic名称和源集群相同
replication.policy.class = org.apache.kafka.connect.mirror.IdentityReplicationPolicy
```

**步骤6** 在二进制软件包目录下，启动MirrorMaker，进行数据同步。

```
./bin/connect-mirror-maker.sh config/connect-mirror-maker.properties
```

**步骤7** （可选）MirrorMaker开启后，如果在源集群上新建了Topic，如需对此Topic进行数据同步，则需重启MirrorMaker，重启步骤参考**步骤6**。也可配置自动同步新增Topic，按

需增加如表11-3所示配置后，无需重启MirrorMaker，即可周期性同步新增Topic。其中，“refresh.topics.interval.seconds”为必选，其他参数根据实际情况选择。

表 11-3 MirrorMaker 配置参数

| 参数名                               | 默认值                                                      | 说明                                       |
|-----------------------------------|----------------------------------------------------------|------------------------------------------|
| sync.topic.configs.enabled        | true                                                     | 是否监控源集群的配置更改                             |
| sync.topic.acls.enabled           | true                                                     | 是否监控源集群ACL的更改                            |
| emit.heartbeats.enabled           | true                                                     | 连接器应定期发出心跳                               |
| emit.heartbeats.interval.seconds  | 5秒                                                       | 心跳频率                                     |
| emit.checkpoints.enabled          | true                                                     | 连接器应定期发出消费端偏移量信息                         |
| emit.checkpoints.interval.seconds | 5秒                                                       | 检查点的频率                                   |
| refresh.topics.enabled            | true                                                     | 连接器应定期检查新主题                              |
| refresh.topics.interval.seconds   | 5秒                                                       | 检查源集群中是否有新主题的频率                          |
| refresh.groups.enabled            | true                                                     | 连接器应定期检查新的消费组                            |
| refresh.groups.interval.seconds   | 5秒                                                       | 检查源集群新的消费组频率                             |
| replication.policy.class          | org.apache.kafka.connect.mirror.DefaultReplicationPolicy | 使用LegacyReplicationPolicy模拟旧版MirrorMaker |
| heartbeats.topic.retention.ms     | 1天                                                       | 首次创建心跳主题时使用                              |
| checkpoints.topic.retention.ms    | 1天                                                       | 首次创建检查点主题时使用                             |
| offset.syncs.topic.retention.ms   | max long                                                 | 首次创建偏移同步主题时使用                            |

----结束

验证数据是否同步

步骤1 在目标集群中查看Topic列表，确认是否有源集群Topic。

## 📖 说明

“replication.policy.class”的默认值为  
“org.apache.kafka.connect.mirror.DefaultReplicationPolicy”，此时目标集群中的Topic名称  
和源集群相比，多了前缀（如A.），这是MirrorMaker为了防止Topic循环备份进行的设置。如果  
想要Topic名称保持一致，请将“replication.policy.class”设置为  
“org.apache.kafka.connect.mirror.IdentityReplicationPolicy”。

**步骤2** 在源集群生产并消费消息，在目标集群查看消费进度，确认数据是否已从源集群同步到了目标集群。

如果目标集群为华为云Kafka实例的话，在分布式消息服务Kafka版控制台的“实例管理 > 消费组管理 > 消费进度”中，查看消费进度。

----结束

## 11.3 使用 Smart Connect 迁移数据

### 11.3.1 开启 Smart Connect

Smart Connect用于Kafka实例和其他云服务（如OBS）之间的数据同步，或者两个Kafka实例之间的数据同步，实现数据的备份或迁移。

Smart Connect的使用流程为：

1. 开启Smart Connect。
2. 创建Smart Connect任务。

本章节主要介绍如何开启Smart Connect。

### 约束与限制

- 开启Smart Connect后，实例需要另外收取用于Smart Connect的代理费用。  
例如：规格为kafka.4u8g.cluster的实例，会另外创建至少两个规格为kafka.4u8g的代理，用于Smart Connect，所以需要收取相应代理费用。
- 单机实例不支持开启Smart Connect。

### 前提条件

- 已[创建Kafka实例](#)，且实例状态为“运行中”。
- “auto.create.groups.enable”已设置为“true”。如果需要修改“auto.create.groups.enable”的取值，请参考[修改Kafka实例配置参数](#)。

### 开启 Smart Connect

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，开启Smart Connect。

- 在待开启Smart Connect的Kafka实例所在行，单击“更多 > 开启Smart Connect”。

- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 开启Smart Connect”。
- 单击Kafka实例名称，进入实例详情页面。在“Smart Connect”后，单击.
- 单击Kafka实例名称，进入实例详情页面。在左侧导航栏单击“Smart Connect”，进入Smart Connect页面。单击“开启Smart Connect”。

**步骤4** 单击，将Smart Connect设置为开启，并设置用于Smart Connect的代理数量，代理数取值范围为2~16个，单击“下一步”。

#### 注意

代理数默认为2个，如果您预估2个Kafka实例间的同步流量比较大，例如大于50MB/s，请设置2个以上的代理数。

**步骤5** 确认“Smart Connect”为开启状态，单击“提交”。

在“实例管理 > 后台任务管理”页面的“即时任务”页签中，当“开启Smart Connect”任务的状态变为“成功”时，表示开启Smart Connect成功。

----结束

## 后续操作

开启Smart Connect后，您可以[配置Kafka间的数据复制](#)和[配置Kafka数据转储至 OBS](#)，实现分布式消息服务Kafka版和其他云服务之间的数据同步。

## 相关文档

如果您希望通过API开启Smart Connect，请参考[开启Smart Connect（按需实例）](#)。

## 11.3.2 配置 Kafka 间的数据复制

本章节介绍如何创建Kafka数据复制的Smart Connect任务，通过Smart Connect任务可以在两个Kafka实例之间，实现数据的单向或双向复制。

## 数据同步规则

Smart Connect任务将源Topic的[配置信息](#)和[消息数据](#)同步到目标Topic，具体同步规则如[表11-4](#)所示。

表 11-4 同步规则

| 同步类型 | 同步规则                                                                                                                                                                                                                                                                                                                                                           |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 配置信息 | <p>分为初始化同步和定时同步。</p> <ul style="list-style-type: none"><li>初始化同步：Smart Connect任务首次向目标Topic同步配置信息时，会先判断是否需要创建对应Topic。如果需要创建，会将源Topic的配置信息同步至新建Topic。如果无需创建，则不会触发配置信息的初始化同步。<br/>初始化同步支持的配置参数，请参见表11-5。<br/>在创建Smart Connect任务时，需指定是否同步消费进度。任务创建成功后，无法修改该配置。</li><li>定时同步：Smart Connect任务启动后，会按照每分钟一次的同步周期，定期向目标Topic同步配置信息。<br/>定时同步支持的配置参数，请参见表11-5。</li></ul> |
| 消息数据 | 源Topic中的消息数据会实时同步到目标Topic中。                                                                                                                                                                                                                                                                                                                                    |

表 11-5 Topic 支持同步的配置参数

| 初始化同步支持的配置参数                                                         | 定时同步支持的配置参数                                                        |
|----------------------------------------------------------------------|--------------------------------------------------------------------|
| 分区数                                                                  | 分区数<br><b>注意</b><br>分区数仅支持增加，不支持减少。如果目标Topic的分区数大于源Topic，则不会同步分区数。 |
| 老化时间                                                                 | -                                                                  |
| 同步落盘                                                                 | -                                                                  |
| 批处理消息最大值                                                             | -                                                                  |
| 副本数<br><b>注意</b><br>目标Topic的副本数由创建Smart Connect任务时指定，而非同步源Topic的副本数。 | -                                                                  |

约束与限制

- 单机实例不支持创建Kafka数据复制的Smart Connect任务。
- 一个实例最多创建18个Smart Connect任务。
- 使用Kafka数据复制时，两个Kafka实例间只能通过内网连接。如果两个Kafka实例处于不同的VPC中，请先打通网络，方法可以参考[使用VPCEP实现跨VPC访问Kafka](#)或者[对等连接](#)。
- Smart Connect任务创建成功后，不支持修改任务参数。
- 2022年7月1日前已开启Smart Connect的实例，如果没有Kafka数据复制，请先[关闭Smart Connect](#)，然后再重新开启Smart Connect。

- 确保目标Kafka实例Topic的“批处理消息最大值”大于等于524288字节，否则会导致数据无法同步。如果目标Kafka实例没有创建Topic，在数据同步时会自动创建Topic，此Topic的“批处理消息最大值”和源Kafka实例Topic相同，此时需要确保源Kafka实例Topic的“批处理消息最大值”大于等于524288字节。修改“批处理消息最大值”的方法请参考[修改Kafka Topic配置](#)。
- Smart Connect任务的正常运行依赖于源端和目标端的运行环境，包括服务状态、网络连通性、访问权限和配置信息等。如果运行环境出现异常，Smart Connect任务将无法保证数据同步正常。
- Smart Connect任务具备断点续传功能，一般情况下可保证数据不丢失。但需注意，如果同步中断时间过长，源端消息已老化，此时会导致数据丢失。

## 前提条件

- 已[开启Smart Connect](#)。
- 已创建Kafka实例，且实例状态为“运行中”。

## 配置 Kafka 间的数据复制

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 单击“创建Smart Connect任务”，进入“创建smart connect”页面。

**步骤6** 在“connect任务名称”中，输入Smart Connect任务的名称，用于区分不同的Smart Connect任务。任务名称需要符合命名规则：长度为4~64个字符，只能由英文字母、数字、中划线、下划线组成。

**步骤7** 在“预置类型”中，选择“Kafka数据复制”。

**步骤8** 在“立即执行”中，选择任务创建后是否立即执行任务，默认开启立即执行。如果关闭，在需要执行此任务时，您可以在Smart Connect任务列表中手动开启任务。

**步骤9** 在“当前kafka”区域，设置实例别名。实例别名需要符合命名规则：长度为1~20个字符，只能由英文字母、数字、中划线、下划线组成。

实例别名用于以下两个场景中：

- 开启“重命名Topic”，且“同步方式”为“推送”/“双向”时，当前Kafka实例的别名作为前缀添加到对端Kafka实例的Topic名称前，形成Topic新的名称。例如当前Kafka实例别名为A，对端Kafka实例的Topic名称为test，重命名后的Topic为A.test。
- Kafka数据复制的Smart Connect任务创建成功后，当前Kafka实例会自动创建“mm2-offset-syncs.对端Kafka实例别名.internal”的Topic。如果Smart Connect任务开启了“同步消费进度”功能，且“同步方式”为“拉取”/“双向”时，当前Kafka实例还会自动创建“对端Kafka实例别名.checkpoints.internal”的Topic。这两个Topic用于存储内部数据，如果删除，会导致同步数据失败。

**步骤10** 在“对端kafka”区域，设置以下参数。

表 11-6 对端 Kafka 实例参数说明

| 参数      | 参数说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 实例别名    | <p>设置实例别名，实例别名需要符合命名规则：长度为1~20个字符，只能由英文字母、数字、中划线、下划线组成。</p> <p>实例别名用于以下两个场景中：</p> <ul style="list-style-type: none"> <li>开启“重命名Topic”，且“同步方式”为“拉取”/“双向”时，对端Kafka实例的别名作为前缀添加到当前Kafka实例的Topic名称前，形成Topic新的名称。例如对端Kafka实例别名为B，当前Kafka实例的Topic名称为test01，重命名后的Topic为B.test01。</li> <li>Kafka数据复制的Smart Connect任务创建成功后，如果Smart Connect任务开启了“同步消费进度”功能，且“同步方式”为“推送”/“双向”时，对端Kafka实例会自动创建“<i>当前Kafka实例别名.checkpoints.internal</i>”的Topic。此Topic用于存储内部数据，如果删除，会导致同步数据失败。</li> </ul> |
| 配置类型    | <p>支持以下两种配置类型：</p> <ul style="list-style-type: none"> <li>Kafka地址：输入Kafka实例的连接信息。<b>对端Kafka实例和当前Kafka实例处于不同的VPC下时，请选择此配置类型。</b></li> <li>实例名称：选择已创建的Kafka实例。<b>对端Kafka实例和当前Kafka实例处于相同的VPC下时，建议选择此配置类型。</b></li> </ul>                                                                                                                                                                                                                                                     |
| 实例名称    | <p>“配置类型”为“实例名称”，且对端Kafka实例和当前Kafka实例处于相同的VPC下时，需要设置。</p> <p>在下拉列表中，选择已创建的Kafka实例。</p>                                                                                                                                                                                                                                                                                                                                                                                     |
| Kafka地址 | <p>“配置类型”为“Kafka地址”时，需要设置。</p> <p>输入Kafka实例的连接地址和端口号。</p> <p>使用Kafka数据复制时，两个Kafka实例间只能通过内网连接。如果两个Kafka实例处于不同的VPC中，请先打通网络，方法可以参考<a href="#">使用VPCEP实现跨VPC访问Kafka</a>或者<a href="#">对等连接</a>。</p>                                                                                                                                                                                                                                                                             |
| 认证方式    | <p>支持以下认证方式：</p> <ul style="list-style-type: none"> <li>SASL_SSL：表示实例已开启SASL_SSL认证，客户端连接Kafka实例时采用SASL认证，数据通过SSL证书进行加密传输。</li> <li>SASL_PLAINTEXT：表示实例开启SASL_PLAINTEXT认证，客户端连接Kafka实例时采用SASL认证，数据通过明文传输。</li> <li>PLAINTEXT：表示实例未开启认证。</li> </ul>                                                                                                                                                                                                                          |

| 参数   | 参数说明                                                                                                                                                                                       |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 认证机制 | “认证方式”为“SASL_SSL” / “SASL_PLAINTEXT”时，需要设置。 <ul style="list-style-type: none"><li>• SCRAM-SHA-512：采用哈希算法对用户名与密码生成凭证，进行身份校验的安全认证机制，比PLAIN机制安全性更高。</li><li>• PLAIN：一种简单的用户名密码校验机制。</li></ul> |
| 用户名  | “认证方式”为“SASL_SSL” / “SASL_PLAINTEXT”时，需要设置。<br>首次开启密文接入时设置的用户名，或者创建用户时设置的用户名。                                                                                                              |
| 密码   | “认证方式”为“SASL_SSL” / “SASL_PLAINTEXT”时，需要设置。<br>首次开启密文接入时设置的密码，或者创建用户时设置的密码。                                                                                                                |

**Smart Connect任务创建成功后，如果您修改了对端实例的认证方式、认证机制或者密码，会导致同步任务失败。您需要删除当前Smart Connect任务，然后重新创建新的Smart Connect任务。**

**步骤11** 在“规则配置”区域，设置以下参数。

表 11-7 复制数据规则参数说明

| 参数     | 参数说明                                                                                                                                                                                                                                                                             |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 同步方式   | 支持以下三种同步方式： <ul style="list-style-type: none"><li>• 拉取：把对端Kafka实例数据复制到当前Kafka实例中。</li><li>• 推送：把当前Kafka实例数据复制到对端Kafka实例中。</li><li>• 双向：两端Kafka实例数据进行双向复制。</li></ul>                                                                                                              |
| Topics | 设置需要进行数据复制的Topic。 <ul style="list-style-type: none"><li>• 正则表达式：通过正则表达式匹配Topic。</li><li>• 输入/选择：输入Topic名称，如果需要输入多个Topic名称，先输入一个Topic名称，按“Enter”，然后再输入下一个，按“Enter”，依次输入。或者在下拉列表中，选择Topic。<b>最多输入/选择20个Topic。</b></li></ul> Topic名称以internal结尾时（例如：topic.internal），此Topic的数据不会被同步。 |
| 任务数    | 数据复制的任务数。默认值为2，建议保持默认值。<br>如果“同步方式”为“双向”，实际任务数=设置的任务数*2。                                                                                                                                                                                                                         |

| 参数         | 参数说明                                                                                                                                                                                                                                                                                                                                                                                               |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 重命名Topic   | <p>在目标Topic名称前添加源端Kafka实例的别名，形成目标Topic新的名称。例如源端实例别名为A，目标Topic名称为test，重命名后的目标Topic为A.test。</p> <p>两端实例数据双向复制时，开启“重命名Topic”，防止循环复制。</p>                                                                                                                                                                                                                                                              |
| 添加来源header | <p>目标Topic接收复制的消息，此消息header中包含消息来源。</p> <p>两端实例数据双向复制时，默认开启“添加来源header”，防止循环复制。</p>                                                                                                                                                                                                                                                                                                                |
| 同步消费进度     | <p>开启“同步消费进度”后，将消费者消费进度同步到目标Kafka实例。</p> <p>开启“同步消费进度”后，您需要注意以下几点：</p> <ul style="list-style-type: none"> <li>源端Kafka实例和目标端Kafka实例不能同时消费，否则会导致同步的消费进度异常。</li> <li>同步消费进度的频率为每分钟一次，因此会导致目标端的消费进度可能会略小于源端，造成部分消息被重复消费，所以需要消费者客户端业务逻辑兼容重复消费的场景。</li> <li>从源端同步的offset与目标端的offset并非一致关系，而是映射关系，如果消费进度由消费者客户端自行维护，消费者客户端从消费源端Kafka实例变为消费目标端Kafka实例后，不向目标端Kafka实例获取消费进度，可能会导致offset错误或消费进度重置。</li> </ul> |
| 副本数        | <p>在对端实例中自动创建Topic时，指定Topic的副本数，此参数值不能超过对端实例的代理数。</p> <p>如果对端实例中设置了“default.replication.factor”，此参数的优先级高于“default.replication.factor”。</p>                                                                                                                                                                                                                                                         |
| 启动偏移量      | <p>支持两种偏移量：</p> <ul style="list-style-type: none"> <li>最早：最小偏移量，即获取最早的数据。</li> <li>最新：最大偏移量，即获取最新的数据。</li> </ul>                                                                                                                                                                                                                                                                                   |
| 压缩算法       | 复制消息所使用的压缩算法。                                                                                                                                                                                                                                                                                                                                                                                      |
| topic映射    | <p>通过Topic映射，您可以自定义目标端Topic名称。</p> <p>最多新增20个Topic映射。不能同时设置“重命名Topic”和“topic映射”。</p>                                                                                                                                                                                                                                                                                                               |

配置复制数据规则时需要注意以下几点：

- 创建双向数据复制任务时，为了防止循环复制，控制台限定必须开启“重命名Topic”或者“添加来源header”。如果您在两个实例间，对同一个Topic分别创建拉取和推送的任务，即形成双向数据复制，且两个任务都未开启“重命名Topic”和“添加来源header”，此时会导致数据循环复制。

- 如果创建两个或以上配置完全相同的任务，即重复创建任务，且任务已开启“同步消费进度”，此时会导致数据重复复制，且目标Topic消费进度异常。

图 11-2 复制数据规则配置

规则配置

同步方向

拉取

推送

双向

把对端Kafka实例数据复制到当前Kafka实例中

Topics

正则表达式

输入/选择

请输入标准正则表达式，如.\*

任务数

-

2

+

重命名Topic

复制的目标topic添加前缀，防止循环复制，如预览效果所示

添加来源header

在复制的消息header中标记消息的来源，防止循环复制

同步消费进度

将消费者消费进度同步到目标kafka

副本数

-

3

+

自动创建的topic副本数，不能超过目标端的kafka broker数量

启动偏移量

最早

最新

最小偏移量，即读取最早的数据。

压缩算法

none

gzip

snappy

lz4

zstd

topic映射

源端

目标端

操作

上限 20 个，可再创建 20 个。

效果预览

A

{topic\_name}

B

{topic\_name}

- 步骤12** （可选）在页面右下角单击“开始检测”，测试两端Kafka实例的连通性。
- 显示“连通性检测成功”时，表示两端Kafka实例可以正常连接。
- 步骤13** 单击“立即创建”，跳转到Smart Connect任务列表页面，页面右上角显示“创建xxx任务成功”。

Kafka数据复制的Smart Connect任务创建成功后，Kafka会自动创建以下Topic。

- 当前Kafka实例会自动创建“mm2-offset-syncs.对端Kafka实例别名.internal”的Topic。如果Smart Connect任务开启了“同步消费进度”功能，且“同步方式”为“拉取”/“双向”时，当前Kafka实例还会自动创建“对端Kafka实例别名.checkpoints.internal”的Topic。这两个Topic用于存储内部数据，如果删除，会导致同步数据失败。
- 如果Smart Connect任务开启了“同步消费进度”功能，“同步方式”为“推送”/“双向”时，对端Kafka实例会自动创建“当前Kafka实例别

名.checkpoints.internal”的Topic。此Topic用于存储内部数据，如果删除，会导致同步数据失败。

----结束

## 相关文档

如果您希望通过API配置Kafka间的数据复制，请参考[创建Smart Connect任务](#)。

### 11.3.3 配置 Kafka 数据转储至 OBS

本章节介绍如何创建转储的Smart Connect任务，通过Smart Connect任务可以将Kafka实例中的数据转储至OBS中，实现消息数据的备份。

源Kafka实例中的数据会实时同步到转储文件中。

## 约束与限制

- 单机实例不支持创建转储的Smart Connect任务。
- 一个实例最多创建18个Smart Connect任务。
- Smart Connect任务创建成功后，不支持修改任务参数。

## 前提条件

- 已[开启Smart Connect](#)。
- 已创建Kafka实例，且实例状态为“运行中”。
- 已创建的OBS桶和Kafka实例需要在同一个区域。

## 配置 Kafka 数据转储至 OBS

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 单击“创建Smart Connect任务”，进入“创建smart connect”页面。

**步骤6** 在“connect任务名称”中，输入Smart Connect任务的名称，用于区分不同的Smart Connect任务。任务名称需要符合命名规则：长度为4~64个字符，只能由英文字母、数字、中划线、下划线组成。

**步骤7** 在“预置类型”中，选择“转储”。

**步骤8** 在“立即执行”中，选择任务创建后是否立即执行任务，默认开启立即执行。如果关闭，在需要执行此任务时，您可以在Smart Connect任务列表中手动开启任务。

**步骤9** 在“源端配置”中，保持默认设置。

**步骤10** 在“Topic配置”中，设置以下参数。

表 11-8 Topic 配置参数说明

| 参数    | 参数说明                                                           |
|-------|----------------------------------------------------------------|
| 正则表达式 | 表示以正则表达式的方式订阅Topic，并进行转储。                                      |
| 输入/选择 | 输入/选择待转储的Topic名称，支持选择多个Topic，以逗号“,”分隔。 <b>最多输入/选择20个Topic。</b> |

**步骤11** 在“目标端配置”中，设置以下参数。

表 11-9 目标端参数说明

| 参数      | 参数说明                                                                                                                                                                                                 |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 转储启动偏移量 | 支持以下两种转储启动偏移量： <ul style="list-style-type: none"> <li>最早：最小偏移量，即读取最早的数据。</li> <li>最新：最大偏移量，即获取最新的数据。</li> </ul>                                                                                      |
| 数据转储周期  | 两次转储的时间间隔，单位：秒，默认配置为300秒。<br>若某个时间段内无数据，则此时间段不会生成打包文件。                                                                                                                                               |
| AK      | 访问密钥ID。<br>获取AK的方法，具体请参见 <a href="#">新增访问密钥</a> 。                                                                                                                                                    |
| SK      | 与访问密钥ID结合使用的密钥。<br>获取SK的方法，具体请参见 <a href="#">新增访问密钥</a> 。                                                                                                                                            |
| 转储地址    | 设置存储Topic数据的OBS桶的名称。 <ul style="list-style-type: none"> <li>选择：在下拉列表中选择已创建的OBS桶名称，或者单击“创建转储地址”，新创建一个OBS桶。</li> <li>输入：输入已创建的OBS桶名称，或者单击“创建转储地址”，新创建一个OBS桶。<b>输入的OBS桶必须和Kafka实例在同一个区域。</b></li> </ul> |
| 转储目录    | OBS中存储Topic的目录，多级目录可以用“/”进行分隔。                                                                                                                                                                       |
| 时间目录格式  | 数据将存储在转储目录下的时间目录中，时间目录是按时间格式作为层级的目录。例如，当选择的时间目录格式精确到日時，存储目录为：“桶名称/转储目录/转储Topic名称/年/月/日”。                                                                                                             |
| 记录分隔符   | 在下拉列表中选择一种分隔符，分隔不同的转储记录。                                                                                                                                                                             |
| 存储Key   | 是否转储Key，开启表示转储Key，关闭表示不转储Key。                                                                                                                                                                        |

#### 说明

- 消息进行转储时，不支持以Key作为文件名。
- OBS转储目录格式为：桶名称/转储目录/转储Topic名称/时间目录格式，例如：obs-kafka/smartconnect/topic01/2025/01/01。
- OBS转储文件名称格式为：转储Topic名称+分区编号+转储起始offset，例如：topic01+0+0000000004。
- OBS转储文件切分规则为：以配置的数据转储周期为基准，每个数据转储周期产生一个文件。

**步骤12** 单击“立即创建”，跳转到Smart Connect任务列表页面，页面右上角显示“创建xxx任务成功”。

----结束

## 相关文档

如果您希望通过API配置Kafka数据转储至OBS，请参考[创建Smart Connect任务](#)。

## 11.3.4 管理 Smart Connect 任务

在创建Smart Connect任务后，可以在控制台对任务进行管理操作，包括查看任务、删除任务、启动/暂停任务、重启任务等。

## 约束与限制

单机实例不支持管理Smart Connect任务。

## 前提条件

已[创建Smart Connect任务](#)。

## 查看 Smart Connect 任务

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 单击Smart Connect任务名称，进入Smart Connect任务详情页面。

**步骤6** 查看Smart Connect任务的基本信息、源端信息和目标端信息。

#### 说明

如果Smart Connect任务没有配置源端信息/目标端信息，在Smart Connect任务详情页面不显示源端信息/目标端信息。

----结束

## 删除 Smart Connect 任务

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 在待删除的Smart Connect任务所在行，单击“删除”，弹出“删除任务”对话框。

**步骤6** 单击“确定”，完成Smart Connect任务的删除。

已删除的Smart Connect任务不再显示在Smart Connect任务列表中，表明Smart Connect任务已成功删除。

----结束

## 启动/暂停 Smart Connect 任务

暂停任务后，Kafka实例的数据将不会再同步到另一个Kafka实例或者其他云服务中。

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 执行以下操作，启动/暂停Smart Connect任务。

- 启动：在待启动的Smart Connect任务所在行，单击“启动”。
- 暂停：在待暂停的Smart Connect任务所在行，单击“暂停”，弹出“暂停任务”对话框，单击“确定”。

当Smart Connect任务列表页面中该任务的状态变为“运行中”/“已暂停”时，表示已成功启动/暂停Smart Connect任务。

----结束

## 重启 Smart Connect 任务

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击，选择Kafka实例所在的区域。

**步骤3** 单击Kafka实例名称，进入实例详情页面。

**步骤4** 在左侧导航栏单击“Smart Connect”，进入Smart Connect任务列表页面。

**步骤5** 在待重启的Smart Connect任务所在行，单击“重启”，弹出“重启任务”对话框。

重启Smart Connect任务前，请注意以下两点：

- Smart Connect任务创建后，修改了源端或者目标端的参数，可能会导致重启失败。

- 重启Smart Connect任务会重置同步进度，并重新开始同步任务。

**步骤6** 单击“确定”，完成Smart Connect任务的重启。

当页面上方显示“成功重启任务xxx”时，表示成功重启Smart Connect任务。

----结束

## 相关文档

- 如果您希望通过API查看Smart Connect任务详情，请参考[查询Smart Connect任务详情](#)。
- 如果您希望通过API删除Smart Connect任务，请参考[删除Smart Connect任务](#)。
- 如果您希望通过API暂停Smart Connect任务，请参考[暂停Smart Connect任务](#)。

## 11.3.5 关闭 Smart Connect

如果不再使用Smart Connect相关功能，您可以关闭Smart Connect，释放资源。

关闭Smart Connect不会影响业务。

## 约束与限制

- 关闭Smart Connect后，实例会停止收取用于Smart Connect的代理费用，并自动删除用于Smart Connect的代理。
- 关闭Smart Connect后，再重新开启Smart Connect，已删除的Smart Connect任务无法找回，需要重新创建。
- 单机实例不支持关闭Smart Connect。

## 前提条件

- 已创建Kafka实例，且实例状态为“运行中”。
- 关闭Smart Connect前，请先[删除所有的Smart Connect任务](#)，否则无法关闭Smart Connect。此操作是为了防止关闭Smart Connect导致正在运行的Smart Connect任务丢失。

## 关闭 Smart Connect

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，关闭Smart Connect。

- 在待关闭Smart Connect的Kafka实例所在行，单击“更多 > 关闭Smart Connect”。
- 单击Kafka实例名称，进入实例详情页面。单击右上角的“... > 关闭Smart Connect”。

**步骤4** 单击 ，将Smart Connect设置为关闭，单击“下一步”。

**步骤5** 确认“Smart Connect”为关闭状态，单击“提交”。

在“实例管理 > 后台任务管理”页面的“即时任务”页签中，当“关闭Smart Connect”任务的状态变为“成功”时，表示关闭Smart Connect成功。

----结束

## 相关文档

如果您希望通过API关闭Smart Connect，请参考[关闭Smart Connect（按需实例）](#)。

# 12 测试实例性能

## 12.1 测试 Kafka 生产速率和 CPU 消耗

本章节从批处理大小、是否跨AZ生产、副本数、同步/异步复制的维度进行分布式消息服务Kafka版的性能测试，对比客户端消息生产速率和服务端CPU消耗，得出性能测试结果。

- 测试场景一（批处理大小）：相同的Kafka实例和Topic，不同的消息大小
- 测试场景二（是否跨AZ生产）：相同的Kafka实例和Topic，生产客户端和服务端在不同的AZ中
- 测试场景三（副本数）：相同的Kafka实例，不同的Topic副本数
- 测试场景四（同步/异步复制）：相同的Kafka实例，不同复制机制的Topic

表 12-1 测试参数

| 分区数 | 副本数 | 是否同步复制 | batch.size | 是否跨AZ生产 |
|-----|-----|--------|------------|---------|
| 3   | 1   | 否      | 1KB        | 否       |
| 3   | 1   | 否      | 16KB       | 否       |
| 3   | 1   | 否      | 1KB        | 是       |
| 3   | 3   | 是      | 1KB        | 否       |
| 3   | 3   | 否      | 1KB        | 否       |

### 测试环境

进行性能测试前，您需要先构建如下的测试环境：

1. 购买一个Kafka实例，参数信息如下，其他参数保持默认，购买方法请参考[购买Kafka实例](#)。

表 12-2

| 参数名称    | 示例取值               |
|---------|--------------------|
| 区域      | 华北-北京四             |
| 可用区     | 可用区1               |
| 版本      | 2.7                |
| 部署架构    | 集群                 |
| 代理规格    | kafka.2u4g.cluster |
| 代理数量    | 3                  |
| 单代理存储空间 | 超高I/O，200GB        |
| 虚拟私有云   | 选择虚拟私有云            |
| 子网      | 选择子网               |
| 安全组     | 选择安全组              |
| 访问方式    | 保持默认               |
| 实例名称    | kafka-test         |
| 企业项目    | default            |

购买完成后，在实例详情页获取Kafka实例的内网明文连接地址。



2. 在购买的Kafka实例中，创建如下参数的3个Topic，具体步骤请参考[创建Kafka Topic](#)。
- Topic-01：3分区1副本，异步复制

- Topic-02：3分区3副本，异步复制

- Topic-03：3分区3副本，同步复制
3. 获取测试工具。
- 获取[Kafka命令行工具2.7.2版本](#)。

#### 4. 购买客户端服务器。

购买如下参数的2台ECS服务器，具体步骤请参考[购买弹性云服务器](#)。

- 区域、可用区、虚拟私有云、子网、安全组与Kafka实例保持一致，规格为4U8G，Linux系统的ECS。
- 区域、虚拟私有云、子网、安全组与Kafka实例保持一致，“可用区”为“可用区2”，规格为4U8G，Linux系统的ECS。

购买完成ECS后，需要在ECS中完成以下配置：

- 安装[Java JDK](#)，并配置JAVA\_HOME与PATH环境变量。  
export JAVA\_HOME=/root/jdk1.8.0\_231  
export PATH=\$JAVA\_HOME/bin:\$PATH
- 下载[Kafka命令行工具2.7.2版本](#)，并解压。  
tar -zxf kafka\_2.12-2.7.2.tgz

## 测试脚本

```
./kafka-producer-perf-test.sh --producer-props bootstrap.servers={连接地址} acks=1 batch.size={batch.size}
linger.ms=0 --topic {Topic名称} --num-records {num-records} --record-size 1024 --throughput 102400
```

表 12-3 脚本参数说明

| 参数名称              | 说明                                        |
|-------------------|-------------------------------------------|
| bootstrap.servers | <a href="#">购买Kafka实例</a> 中获取的Kafka实例的地址。 |
| acks              | 消息主从同步策略，acks=1表示异步复制消息，acks=-1表示同步复制消息。  |
| batch.size        | 每次批量发送消息的大小（单位为字节）。                       |
| linger.ms         | 两次发送时间间隔。                                 |
| topic             | <a href="#">创建Topic</a> 中设置的Topic名称。      |
| num-records       | 总共需要发送的消息数。                               |
| record-size       | 每条消息的大小。                                  |
| throughput        | 每秒发送的消息数。                                 |

## 测试步骤

测试场景一：批处理大小

1. 登录客户端服务器，进入“kafka\_2.12-2.7.2/bin”目录下，执行以下脚本。

**batch.size=1KB**，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-01 --num-records 8000000 --record-size 1024 --throughput 102400
```

执行结果如下：

```
8000000 records sent, 34128.673632 records/sec (33.33 MB/sec), 879.91 ms avg latency, 4102.00 ms
max latency, 697 ms 50th, 2524 ms 95th, 2888 ms 99th, 4012 ms 99.9th.
```

客户端消息生产速率=34128

**batch.size=16KB**，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=16384
linger.ms=0 --topic Topic-01 --num-records 100000000 --record-size 1024 --throughput 102400
```

执行结果如下：

100000000 records sent, 102399.318430 records/sec (100.00 MB/sec), 4.72 ms avg latency, 914.00 ms max latency, 1 ms 50th, 5 ms 95th, 162 ms 99th, 398 ms 99.9th.

客户端消息生产速率=102399

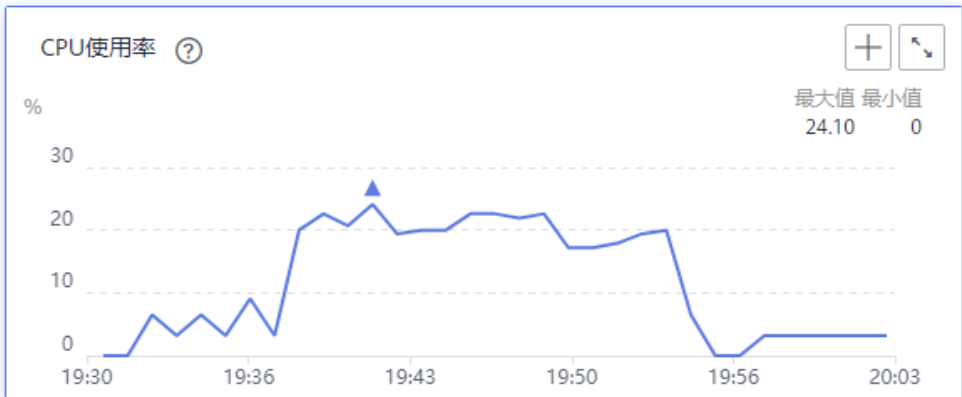
- 2. 登录[Kafka控制台](#)。
- 3. 单击测试实例名称，进入实例详情页。
- 4. 在左侧导航栏单击“监控 > 监控详情”，进入监控页面。
- 5. 在“节点”页签，查看服务端节点的CPU使用率。

图 12-1 broker-0 的 CPU 使用率（batch.size=1KB）



CPU消耗=58.10%

图 12-2 broker-0 的 CPU 使用率（batch.size=16KB）



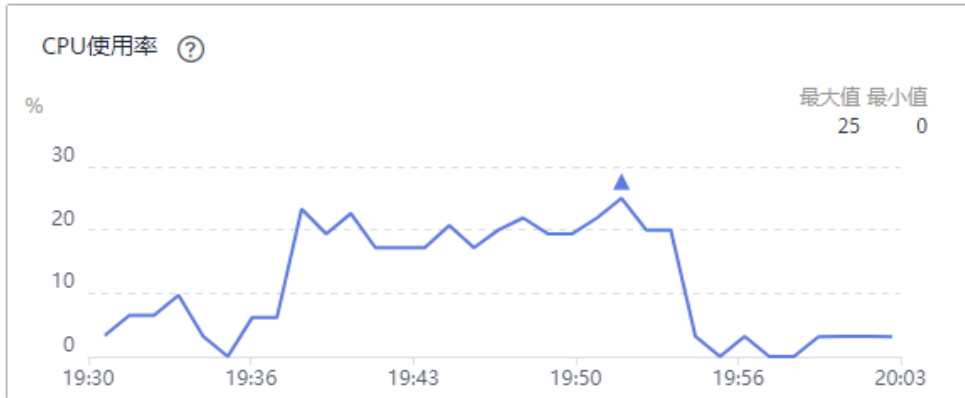
CPU消耗=24.10%

图 12-3 broker-1 的 CPU 使用率 ( batch.size=1KB )



CPU消耗=56.70%

图 12-4 broker-1 的 CPU 使用率 ( batch.size=16KB )



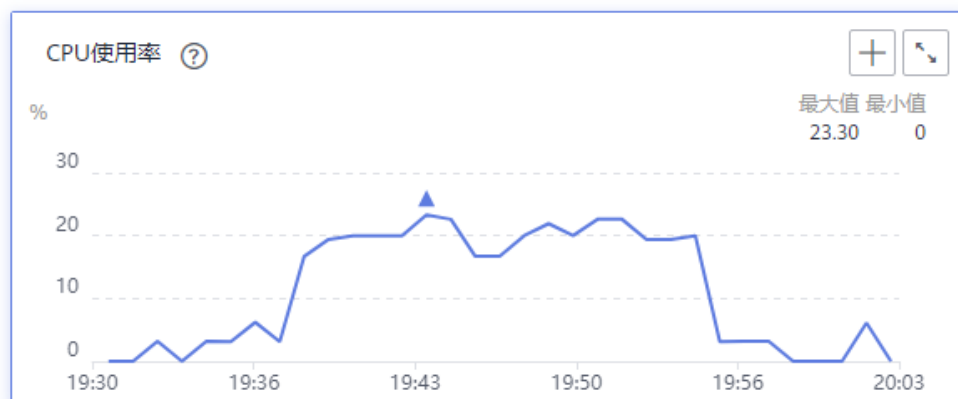
CPU消耗=25%

图 12-5 broker-2 的 CPU 使用率 ( batch.size=1KB )



CPU消耗=53.30%

图 12-6 broker-2 的 CPU 使用率 ( batch.size=16KB )



CPU消耗=23.30%

测试场景二：是否跨AZ生产

1. 登录客户端服务器，进入“kafka\_2.12-2.7.2/bin”目录下，执行以下脚本。

**客户端服务器和实例在相同的AZ中，执行脚本如下：**

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-01 --num-records 8000000 --record-size 1024 --throughput 102400
```

执行结果如下：

```
8000000 records sent, 34128.673632 records/sec (33.33 MB/sec), 879.91 ms avg latency, 4102.00 ms
max latency, 697 ms 50th, 2524 ms 95th, 2888 ms 99th, 4012 ms 99.9th.
```

客户端消息生产速率=34128

**客户端服务器和实例在不同的AZ中，执行脚本如下：**

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-01 --num-records 4000000 --record-size 1024 --throughput 102400
```

执行结果如下：

```
4000000 records sent, 8523.042044 records/sec (8.32 MB/sec), 3506.20 ms avg latency, 11883.00 ms
max latency, 1817 ms 50th, 10621 ms 95th, 11177 ms 99th, 11860 ms 99.9th.
```

客户端消息生产速率=8523

2. 登录[Kafka控制台](#)。
3. 单击测试实例名称，进入实例详情页。
4. 在左侧导航栏单击“监控 > 监控详情”，进入监控页面。
5. 在“节点”页签，查看服务端节点的CPU使用率。

图 12-7 broker-0 的 CPU 使用率（客户端服务器和实例在相同的 AZ 中）



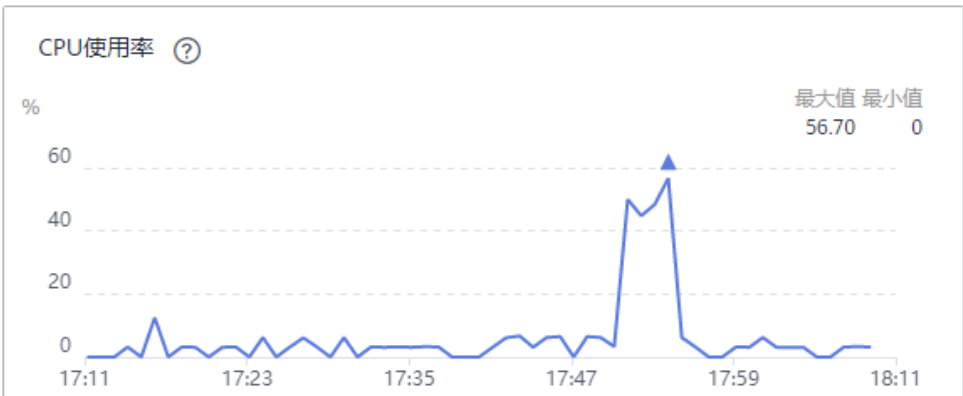
CPU消耗=58.10%

图 12-8 broker-0 的 CPU 使用率（客户端服务器和实例在不同的 AZ 中）



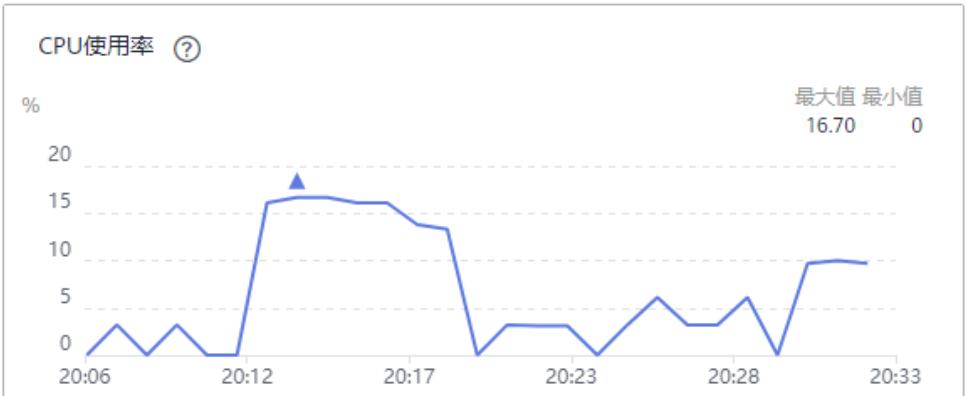
CPU消耗=17.20%

图 12-9 broker-1 的 CPU 使用率（客户端服务器和实例在相同的 AZ 中）



CPU消耗=56.70%

图 12-10 broker-1 的 CPU 使用率（客户端服务器和实例在不同的 AZ 中）



CPU消耗=16.70%

图 12-11 broker-2 的 CPU 使用率（客户端服务器和实例在相同的 AZ 中）



CPU消耗=53.30%

图 12-12 broker-2 的 CPU 使用率（客户端服务器和实例在不同的 AZ 中）



CPU消耗=18.80%

测试场景三：副本数

1. 登录客户端服务器，进入“kafka\_2.12-2.7.2/bin”目录下，执行以下脚本。

**1副本**，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-01 --num-records 8000000 --record-size 1024 --throughput 102400
```

执行结果如下：

```
8000000 records sent, 34128.673632 records/sec (33.33 MB/sec), 879.91 ms avg latency, 4102.00 ms
max latency, 697 ms 50th, 2524 ms 95th, 2888 ms 99th, 4012 ms 99.9th.
```

客户端消息生产速率=34128

**3副本**，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-02 --num-records 4000000 --record-size 1024 --throughput 102400
```

执行结果如下：

```
4000000 records sent, 14468.325219 records/sec (14.13 MB/sec), 2069.99 ms avg latency, 7911.00 ms
max latency, 846 ms 50th, 6190 ms 95th, 6935 ms 99th, 7879 ms 99.9th.
```

客户端消息生产速率=14468

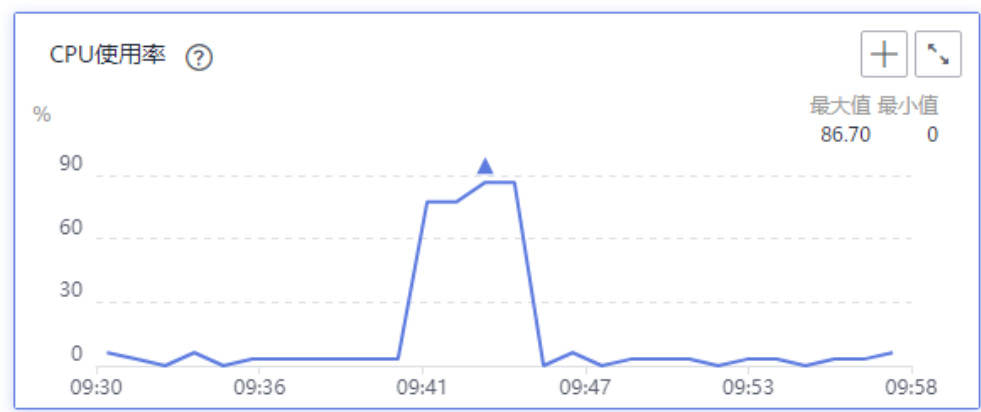
2. 登录[Kafka控制台](#)。  
3. 单击测试实例名称，进入实例详情页。  
4. 在左侧导航栏单击“监控 > 监控详情”，进入监控页面。  
5. 在“节点”页签，查看服务端节点的CPU使用率。

图 12-13 broker-0 的 CPU 使用率（1 副本）



CPU消耗=58.10%

图 12-14 broker-0 的 CPU 使用率（3 副本）



CPU消耗=86.70%

图 12-15 broker-1 的 CPU 使用率（1 副本）



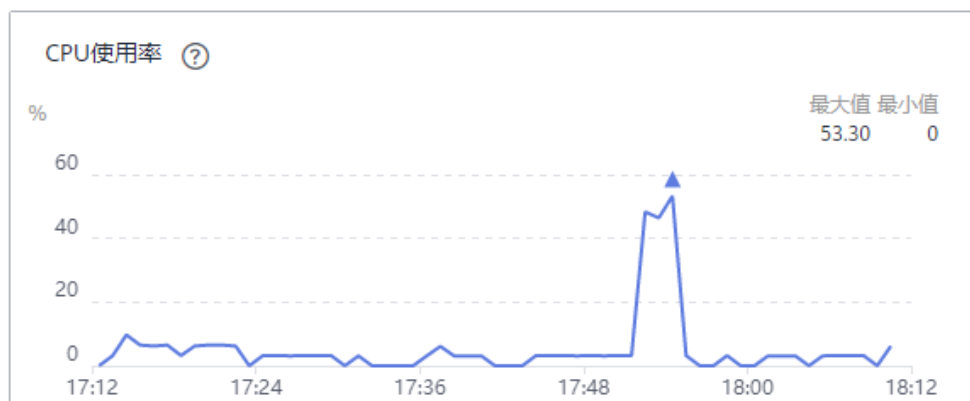
CPU消耗=56.70%

图 12-16 broker-1 的 CPU 使用率（3 副本）



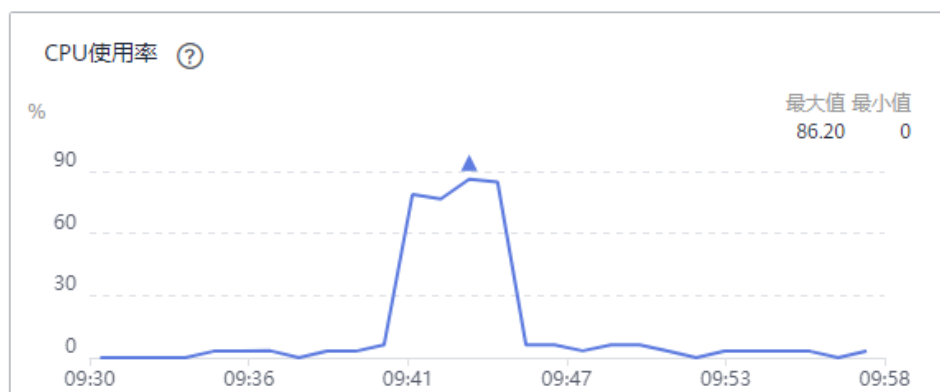
CPU消耗=80.60%

图 12-17 broker-2 的 CPU 使用率（1 副本）



CPU消耗=53.30%

图 12-18 broker-2 的 CPU 使用率（3 副本）



CPU消耗=86.20%

#### 测试场景四：同步/异步复制

1. 登录客户端服务器，进入“kafka\_2.12-2.7.2/bin”目录下，执行以下脚本。

##### 异步复制，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=1 batch.size=1024
linger.ms=0 --topic Topic-02 --num-records 4000000 --record-size 1024 --throughput 102400
```

##### 执行结果如下：

4000000 records sent, 14468.325219 records/sec (14.13 MB/sec), 2069.99 ms avg latency, 7911.00 ms max latency, 846 ms 50th, 6190 ms 95th, 6935 ms 99th, 7879 ms 99.9th.

客户端消息生产速率=14468

##### 同步复制，执行脚本如下：

```
./kafka-producer-perf-test.sh --producer-props
bootstrap.servers=192.168.0.69:9092,192.168.0.42:9092,192.168.0.66:9092 acks=-1 batch.size=1024
linger.ms=0 --topic Topic-03 --num-records 1000000 --record-size 1024 --throughput 102400
```

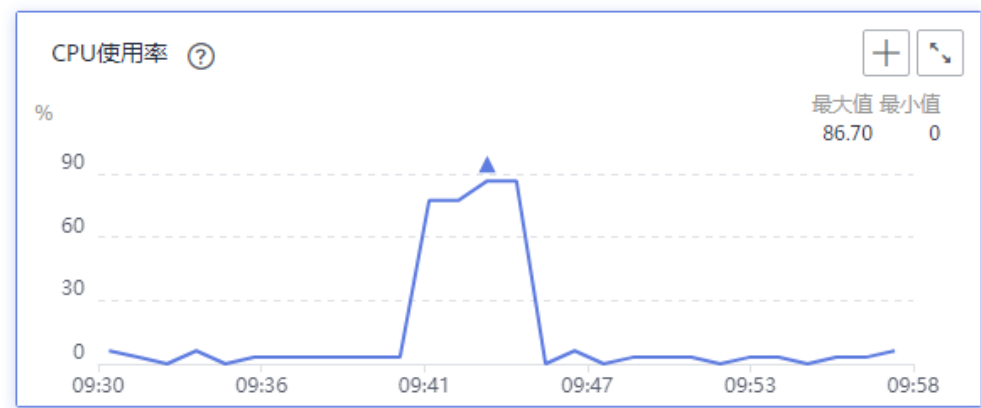
##### 执行结果如下：

1000000 records sent, 3981.937930 records/sec (3.89 MB/sec), 7356.98 ms avg latency, 19013.00 ms max latency, 6423 ms 50th, 14381 ms 95th, 18460 ms 99th, 18975 ms 99.9th.

客户端消息生产速率=3981

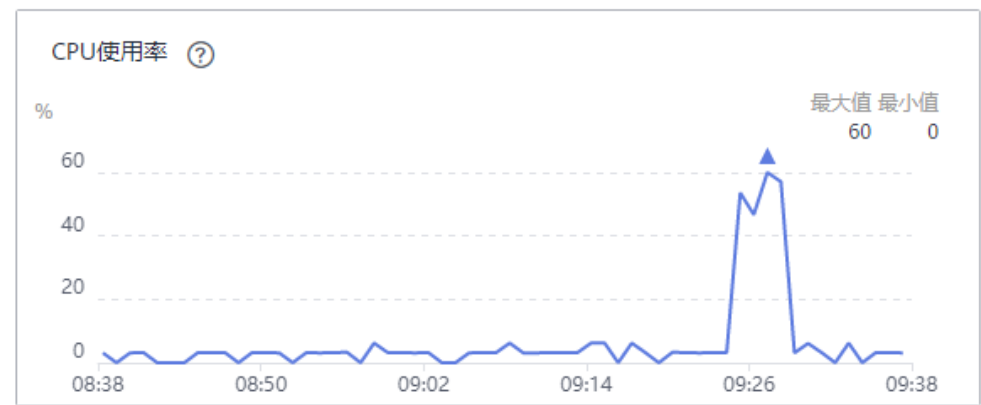
- 2. 登录[Kafka控制台](#)。
- 3. 单击测试实例名称，进入实例详情页。
- 4. 在左侧导航栏单击“监控 > 监控详情”，进入监控页面。
- 5. 在“节点”页面，查看服务端节点的CPU使用率。

图 12-19 broker-0 的 CPU 使用率（异步复制）



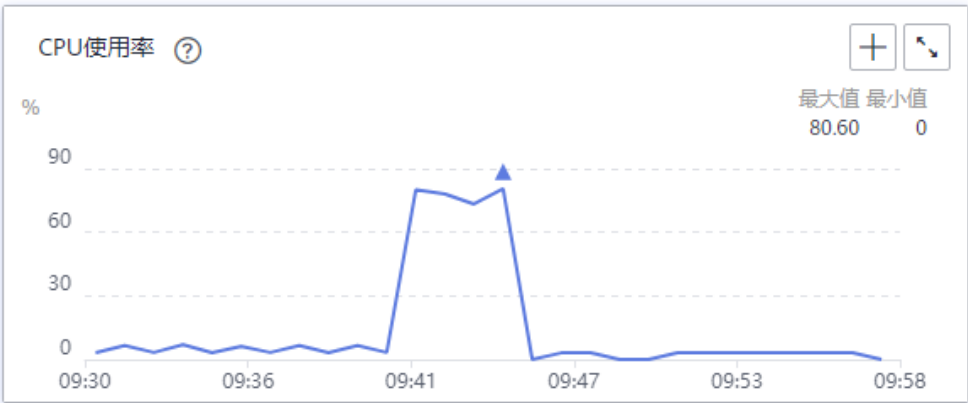
CPU消耗=86.70%

图 12-20 broker-0 的 CPU 使用率（同步复制）



CPU消耗=60%

图 12-21 broker-1 的 CPU 使用率（异步复制）



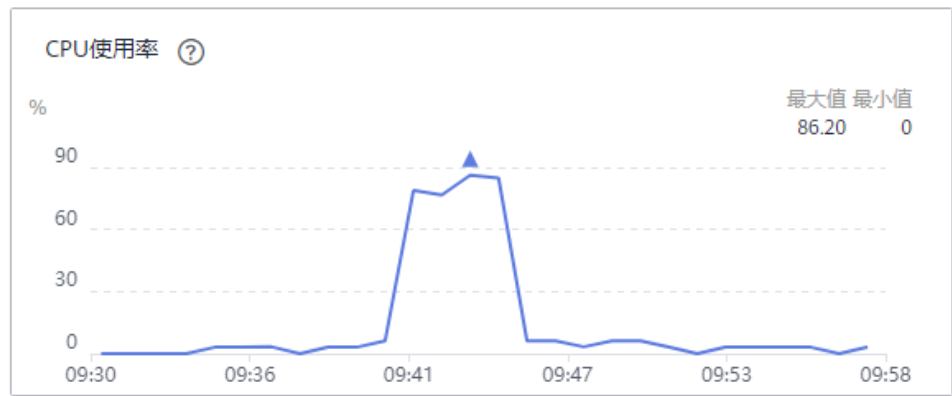
CPU消耗=80.60%

图 12-22 broker-1 的 CPU 使用率（同步复制）



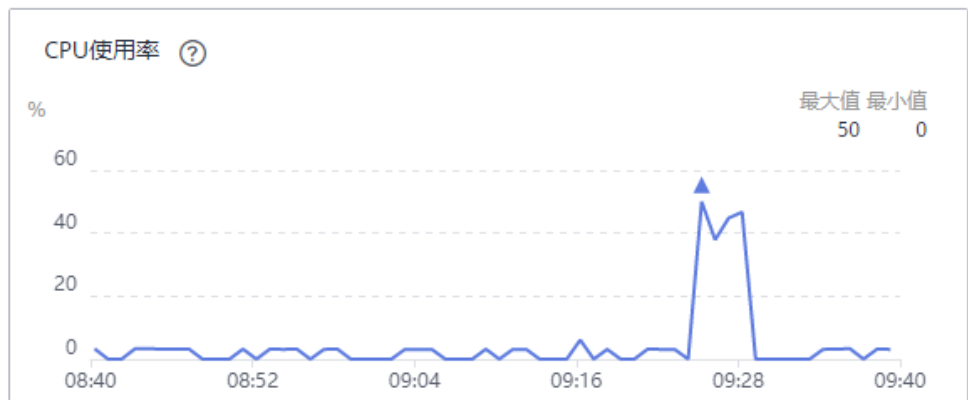
CPU消耗=55.20%

图 12-23 broker-2 的 CPU 使用率（异步复制）



CPU消耗=86.20%

图 12-24 broker-2 的 CPU 使用率（同步复制）



CPU消耗=50%

测试结果

表 12-4 测试结果

| 分区数 | 副本数 | 是否同步复制 | batch.size | 是否跨AZ生产 | 客户端消息生产速率 | 服务端CPU消耗 (broker-0) | 服务端CPU消耗 (broker-1) | 服务端CPU消耗 (broker-2) |
|-----|-----|--------|------------|---------|-----------|---------------------|---------------------|---------------------|
| 3   | 1   | 否      | 1KB        | 否       | 34128     | 58.10%              | 56.70%              | 53.30%              |
| 3   | 1   | 否      | 16KB       | 否       | 102399    | 24.10%              | 25.00%              | 23.30%              |
| 3   | 1   | 否      | 1KB        | 是       | 8523      | 17.20%              | 16.70%              | 18.80%              |
| 3   | 3   | 是      | 1KB        | 否       | 3981      | 60.00%              | 55.20%              | 50.00%              |
| 3   | 3   | 否      | 1KB        | 否       | 14468     | 86.70%              | 80.60%              | 86.20%              |

通过上表的测试结果，得出以下结论，仅供参考：

- 生产请求的batch.size变大16倍时，客户端消息生产速率增加，服务端CPU消耗减少。
- 同AZ生产和跨AZ生产相比，客户端消息生产速率增加，服务端CPU消耗也随之增加。
- 副本从1变成3时，客户端消息生产速率下降较多，服务端CPU消耗增加。
- 异步复制的Topic和同步复制的Topic相比，客户端消息生产速率增加，服务端CPU消耗也随之增加。

12.2 测试 Kafka 实例 TPS

通过以下几个场景，测试不同实例规格的TPS。

- 测试场景一（实例是否开启SASL）：相同的Topic，实例分为开启SASL和未开启SASL
- 测试场景二（同步/异步复制）：相同的实例，不同复制机制的Topic
- 测试场景三（是否同步落盘）：相同的实例，不同落盘机制的Topic
- 测试场景四（不同磁盘类型）：相同的Topic，不同磁盘类型的实例
- 测试场景五（不同分区数）：相同的实例，不同分区数的Topic

测试环境

进行TPS测试前，您需要先构建如下的测试环境：

1. 购买如表12-5所示实例，购买步骤请参考[购买Kafka实例](#)。

表 12-5 实例参数

| 名称       | 代理数量 | 规格                   | 是否开启SASL | 磁盘类型  |
|----------|------|----------------------|----------|-------|
| kafka-01 | 3    | kafka.2u4g.cluster   | 是        | 超高I/O |
| kafka-02 | 3    | kafka.4u8g.cluster   | 是        | 超高I/O |
| kafka-03 | 3    | kafka.8u16g.cluster  | 是        | 超高I/O |
| kafka-04 | 3    | kafka.12u24g.cluster | 是        | 超高I/O |
| kafka-05 | 3    | kafka.16u32g.cluster | 是        | 超高I/O |
| kafka-06 | 3    | kafka.2u4g.cluster   | 否        | 超高I/O |
| kafka-07 | 3    | kafka.4u8g.cluster   | 否        | 超高I/O |
| kafka-08 | 3    | kafka.8u16g.cluster  | 否        | 超高I/O |
| kafka-09 | 3    | kafka.12u24g.cluster | 否        | 超高I/O |
| kafka-10 | 3    | kafka.16u32g.cluster | 否        | 超高I/O |
| kafka-11 | 3    | kafka.2u4g.cluster   | 否        | 高I/O  |
| kafka-12 | 3    | kafka.4u8g.cluster   | 否        | 高I/O  |

| 名称       | 代理数量 | 规格                       | 是否开启 SASL | 磁盘类型 |
|----------|------|--------------------------|-----------|------|
| kafka-13 | 3    | kafka.8u16g.clu<br>ster  | 否         | 高I/O |
| kafka-14 | 3    | kafka.12u24g.cl<br>uster | 否         | 高I/O |
| kafka-15 | 3    | kafka.16u32g.cl<br>uster | 否         | 高I/O |

购买完成后，在实例详情页获取Kafka实例的内网明文连接地址。

连接信息

用户名

--

内网访问

明文接入

☒

内网明文连接地址 IPv4

192.168.1.100:9092, 192.168.1.101:9092, 192.168.1.102:9092

密文接入

☐

公网访问

☐ 已关闭

仅支持开启/关闭IPv4公网访问。

2. 购买实例后，创建如表12-6所示Topic，创建步骤请参考[创建Kafka Topic](#)。

表 12-6 Topic 参数

| 名称       | 是否同步复制 | 是否同步落盘 | 副本数 | 分区数 |
|----------|--------|--------|-----|-----|
| topic-01 | 否      | 否      | 3   | 30  |
| topic-02 | 是      | 否      | 3   | 30  |
| topic-03 | 否      | 是      | 3   | 30  |
| topic-04 | 否      | 否      | 3   | 3   |
| topic-05 | 否      | 否      | 3   | 12  |
| topic-06 | 否      | 否      | 3   | 100 |

3. 获取测试工具。  
获取[Kafka命令行工具2.7.2版本](#)。
4. 购买客户端服务器。  
购买1台ECS服务器（区域、可用区、虚拟私有云、子网、安全组与Kafka实例保持一致，Linux系统），具体步骤请参考[购买弹性云服务器](#)。

购买完成ECS后，需要在ECS中完成以下配置：

- 安装[Java JDK](#)，并配置JAVA\_HOME与PATH环境变量。  
export JAVA\_HOME=/root/jdk1.8.0\_231  
export PATH=\$JAVA\_HOME/bin:\$PATH
- 下载[Kafka命令行工具2.7.2版本](#)，并解压。  
tar -zxf kafka\_2.12-2.7.2.tgz

测试脚本

```
./kafka-producer-perf-test.sh --producer-props bootstrap.servers={连接地址} acks=1 batch.size=16384
linger.ms=10 --topic {Topic名称} --num-records 10000000 --record-size 1024 --throughput -1 --
producer.config ../config/producer.properties
```

表 12-7 脚本参数说明

| 参数名称              | 说明                                         |
|-------------------|--------------------------------------------|
| bootstrap.servers | <a href="#">购买Kafka实例</a> 后，获取的Kafka实例的地址。 |
| acks              | 消息主从同步策略，acks=1表示异步复制消息，acks=-1表示同步复制消息。   |
| batch.size        | 每次批量发送消息的大小（单位为字节）。                        |
| linger.ms         | 两次发送时间间隔。                                  |
| topic             | <a href="#">创建Topic</a> 中设置的Topic名称。       |
| num-records       | 总共需要发送的消息数。                                |
| record-size       | 每条消息的大小。                                   |
| throughput        | 每秒发送的消息数。                                  |

测试结果

以下介绍五个测试场景对应的测试结果。

测试场景如下：

- 测试场景一（实例是否开启SASL）：相同的Topic（30分区、3副本、异步复制、异步落盘），实例分为开启SASL和未开启SASL
- 测试场景二（同步/异步复制）：相同的实例（超高I/O、3个代理、未开启SASL），不同复制机制的Topic，生产者进程数为3时
- 测试场景三（是否同步落盘）：相同的实例（超高I/O、3个代理、未开启SASL），不同落盘机制的Topic
- 测试场景四（不同磁盘类型）：相同的Topic（30分区、3副本、异步复制、异步落盘），不同磁盘类型的实例
- 测试场景五（不同分区数）：相同的实例（超高I/O、3个代理、未开启SASL），不同分区数的Topic

## 测试场景一（实例是否开启 SASL）

表 12-8 测试结果

| 实例规格                 | 磁盘类型  | 代理数量 | TPS（开启 SASL） | TPS（未开启 SASL） |
|----------------------|-------|------|--------------|---------------|
| kafka.2u4g.cluster   | 超高I/O | 3    | 100000       | 280000        |
| kafka.4u8g.cluster   | 超高I/O | 3    | 170000       | 496000        |
| kafka.8u16g.cluster  | 超高I/O | 3    | 200000       | 730000        |
| kafka.12u24g.cluster | 超高I/O | 3    | 320000       | 790000        |
| kafka.16u32g.cluster | 超高I/O | 3    | 360000       | 1000000       |

结论：在Topic相同的情况下，生产消息到规格相同、接入方式不同的Kafka实例，未开启SASL的实例TPS高于开启SASL的实例TPS。

## 测试场景二（同步/异步复制）

表 12-9 测试结果

| 实例规格                 | 是否同步落盘 | 副本数 | 分区数 | TPS（同步复制） | TPS（异步复制） |
|----------------------|--------|-----|-----|-----------|-----------|
| kafka.2u4g.cluster   | 否      | 3   | 30  | 100000    | 280000    |
| kafka.4u8g.cluster   | 否      | 3   | 30  | 230000    | 496000    |
| kafka.8u16g.cluster  | 否      | 3   | 30  | 342000    | 730000    |
| kafka.12u24g.cluster | 否      | 3   | 30  | 383000    | 790000    |
| kafka.16u32g.cluster | 否      | 3   | 30  | 485000    | 1000000   |

结论：生产消息到同一个Kafka实例的不同Topic中，Topic除了复制机制，其他参数相同，异步复制Topic的TPS高于同步复制Topic的TPS。

测试场景三（是否同步落盘）

表 12-10 测试结果

| 实例规格                 | 是否同步复制 | 副本数 | 分区数 | TPS（同步落盘） | TPS（异步落盘） |
|----------------------|--------|-----|-----|-----------|-----------|
| kafka.2u4g.cluster   | 否      | 3   | 30  | 30000     | 280000    |
| kafka.4u8g.cluster   | 否      | 3   | 30  | 32500     | 496000    |
| kafka.8u16g.cluster  | 否      | 3   | 30  | 36100     | 730000    |
| kafka.12u24g.cluster | 否      | 3   | 30  | 37400     | 790000    |
| kafka.16u32g.cluster | 否      | 3   | 30  | 40400     | 1000000   |

结论：生产消息到同一个Kafka实例的不同Topic中，Topic除了落盘机制，其他参数相同，异步落盘Topic的TPS远远高于同步落盘Topic的TPS。

测试场景四（不同磁盘类型）

表 12-11 测试结果

| 实例规格                 | 代理数量 | 是否开启SASL | TPS（高I/O） | TPS（超高I/O） |
|----------------------|------|----------|-----------|------------|
| kafka.2u4g.cluster   | 3    | 否        | 110000    | 250000     |
| kafka.4u8g.cluster   | 3    | 否        | 135000    | 380000     |
| kafka.8u16g.cluster  | 3    | 否        | 213000    | 480000     |
| kafka.12u24g.cluster | 3    | 否        | 240000    | 577000     |
| kafka.16u32g.cluster | 3    | 否        | 280000    | 840000     |

结论：在Topic相同的情况下，生产消息到规格相同、磁盘类型不同的Kafka实例，超高I/O的实例TPS高于高I/O的实例TPS。

测试场景五（不同分区数）

表 12-12 测试结果

| 实例规格                 | 是否同步落盘 | 是否同步复制 | 副本数 | TPS（3分区） | TPS（12分区） | TPS（100分区） |
|----------------------|--------|--------|-----|----------|-----------|------------|
| kafka.2u4g.cluster   | 否      | 否      | 3   | 250000   | 260000    | 250000     |
| kafka.4u8g.cluster   | 否      | 否      | 3   | 330000   | 280000    | 260000     |
| kafka.8u16g.cluster  | 否      | 否      | 3   | 480000   | 410000    | 340000     |
| kafka.12u24g.cluster | 否      | 否      | 3   | 570000   | 750000    | 520000     |
| kafka.16u32g.cluster | 否      | 否      | 3   | 840000   | 1000000   | 630000     |

结论：生产消息到同一个Kafka实例的不同Topic中，Topic除了分区数量，其他参数相同。随着分区数的增加，Kafka的性能通常会随之增加，当分区数达到一定程度后，继续增加分区数可能会导致性能下降。

# 13 申请扩大 Kafka 配额

## 什么是配额？

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

如果当前资源配额限制无法满足使用需要，您可以申请扩大配额。

## 怎样查看我的配额？

1. 登录[Kafka控制台](#)。
  2. 单击管理控制台左上角的 ，选择区域和项目。
  3. 在页面右上角，选择“资源 > 我的配额”。
- 系统进入“服务配额”页面。

图 13-1 我的配额



4. 您可以在“服务配额”页面，查看各项资源的总配额及使用情况。
- 如果当前配额不能满足业务要求，请参考后续操作，申请扩大配额。

## 如何申请扩大配额？

1. 登录[Kafka控制台](#)。
  2. 在页面右上角，选择“资源 > 我的配额”。
- 系统进入“服务配额”页面。
3. 单击“申请扩大配额”。

4. 在“新建工单”页面，根据您的需求，填写相关参数。  
其中，“问题描述”项请填写需要调整的内容和申请原因。
5. 填写完毕后，勾选协议并单击“提交”。

# 14查看监控指标与配置告警

## 14.1 查看 Kafka 监控数据

云监控对Kafka实例的运行状态进行日常监控，可以通过控制台直观的查看Kafka实例各项监控指标和关键指标的Top数据。

### 约束与限制

在Kafka控制台的“监控 > 监控详情”页面中，下拉框中每次最多可以选择50个资源，即每次最多可以查看50个资源的监控数据。如果资源数量超过50个，建议分批查看。

### 前提条件

已创建Kafka实例，且实例中有可消费的消息。

### 查看 Kafka 监控数据

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 通过以下任意一种方法，查看监控数据。

表 14-1 查看监控数据

| 数据类型   | 查看方法                                                                                                                                                                            |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 所有监控数据 | <ul style="list-style-type: none"><li>在Kafka实例名称后，单击“查看监控数据”。跳转到云监控页面，查看监控数据，数据更新周期为1分钟。</li><li>单击Kafka实例名称，进入实例详情页。在左侧导航栏单击“监控 &gt; 监控详情”，进入监控页面，查看监控数据，数据更新周期为1分钟。</li></ul> |

| 数据类型        | 查看方法                                                                                                                   |
|-------------|------------------------------------------------------------------------------------------------------------------------|
| 指定Topic监控数据 | 单击Kafka实例名称，进入实例详情页。在左侧导航栏单击“实例管理 > Topic管理”，进入Topic列表页面。单击待查看监控数据的Topic名称，进入Topic详情页。在“监控”页签中，查看Topic监控数据，数据更新周期为1分钟。 |
| 指定消费组监控数据   | 单击Kafka实例名称，进入实例详情页。在左侧导航栏单击“实例管理 > 消费组管理”，进入消费组列表页面。单击待查看监控数据的消费组名称，进入消费组详情页。在“监控”页签中，查看消费组监控数据，数据更新周期为1分钟。           |

- Kafka实例支持查看以下监控数据：
- 单机实例：查看实例、节点、主题和消费组的监控数据。
  - 集群实例：查看实例、节点、主题、消费组和Smart Connect的监控数据。
- 结束

查看 Kafka 关键指标 Top 数据

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页。
- 步骤4 在左侧导航栏选择“监控 > 监控概览”，进入监控概览页面。
- 步骤5 在“关键指标TopN排序”区域，查看消费组、主题、节点关键指标的Top 5、Top 10和Top 20数据。

表 14-2 关键指标列表

| 维度  | 关键指标                                                                                                       |
|-----|------------------------------------------------------------------------------------------------------------|
| 消费组 | <ul style="list-style-type: none"><li>• 消息堆积数</li><li>• 消费组已消费消息数</li></ul>                                |
| 主题  | <ul style="list-style-type: none"><li>• 队列消息总数</li><li>• 生产流量</li><li>• 消费流量</li></ul>                     |
| 节点  | <ul style="list-style-type: none"><li>• CPU使用率</li><li>• 网络带宽利用率</li><li>• 磁盘读流量</li><li>• 磁盘写流量</li></ul> |

图 14-1 关键指标 TopN 排序



----结束

查看 Kafka 生产入流量的监控视图

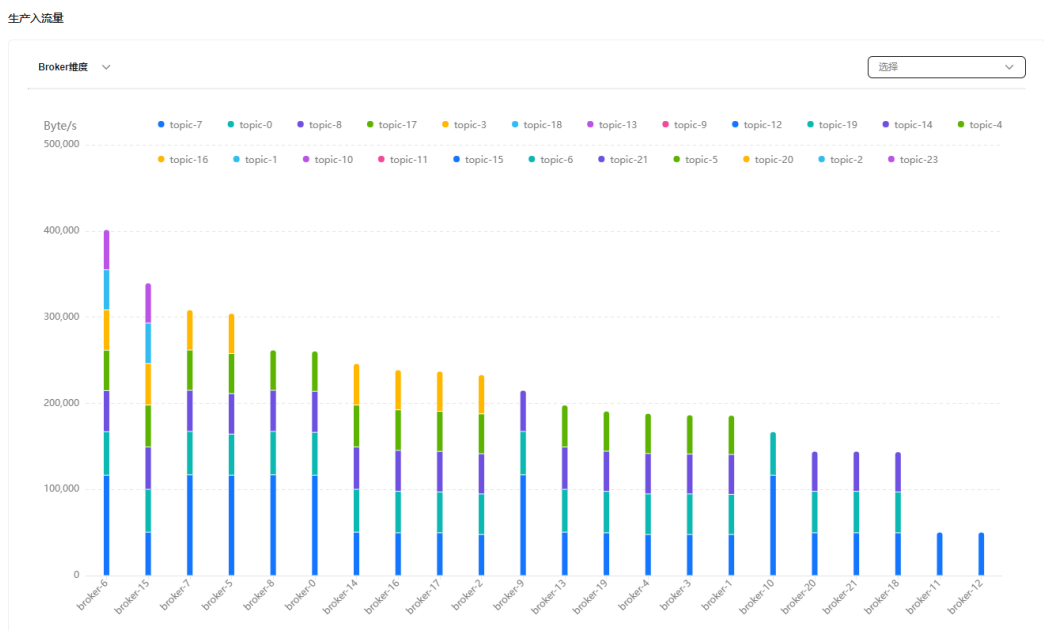
部分存量实例不支持查看生产入流量的监控视图，请联系客服解决。

- 步骤1 登录[Kafka控制台](#)。
- 步骤2 在管理控制台左上角单击 ，选择Kafka实例所在的区域。
- 步骤3 单击Kafka实例名称，进入实例详情页。
- 步骤4 在左侧导航栏选择“监控 > 监控概览”，进入监控概览页面。
- 步骤5 在“关键指标TopN排序”区域的“多维指标”页签中，查看当前Kafka实例生产入流量的监控视图。

表 14-3 生产入流量监控视图说明

| 查看维度     | 说明                                                                                                                                                                                                             |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Topic维度  | <p>按生产入流量从大到小的顺序对Topic进行排序，并以不同颜色显示Topic中不同Broker的生产入流量分布。将鼠标悬停在某个Topic柱形图上，会显示此Topic所在Broker的生产入流量的具体数值。</p> <p>单击右上角的选择框，勾选一个或多个Broker名称，查看Broker生产入流量在Topic中的分布情况。</p> <p><b>Topic维度下最多展示Top 20的数据。</b></p> |
| Broker维度 | <p>按生产入流量从大到小的顺序对Broker进行排序，并以不同颜色显示Broker中不同Topic的生产入流量分布。将鼠标悬停在某个Broker柱形图上，会显示此Broker中所有Topic的生产入流量的具体数值。</p> <p>单击右上角的选择框，勾选一个或多个Topic名称，查看Topic生产入流量在Broker中的分布情况。</p>                                    |

图 14-2 Broker 维度生产入流量监控视图



----结束

相关文档

云监控无法展示Kafka监控数据

14.2 Kafka 支持的监控指标

功能说明

本章节定义了分布式消息服务Kafka版上报云监控服务的监控指标的命名空间，监控指标列表和维度定义，您可以通过云监控服务的管理控制台或API接口来检索Kafka实例产生的监控指标和告警信息，也可以通过分布式消息服务Kafka版控制台提供的“监控详情”页面来检索Kafka实例产生的监控指标。

命名空间

SYS.DMS

实例监控指标

表 14-4 实例支持的监控指标

| 指标 ID                    | 指标名称  | 指标含义                                     | 取值范围         | 单位     | 进制        | 测量对象（维度） | 监控周期（原始指标） |
|--------------------------|-------|------------------------------------------|--------------|--------|-----------|----------|------------|
| current_partitions       | 分区数   | 该指标用于统计Kafka实例中已经使用的分区数量。                | 0~100000     | Count  | 不涉及       | Kafka实例  | 1分钟        |
| current_topics           | 主题数   | 该指标用于统计Kafka实例中已经创建的主题数量。                | 0~100000     | Count  | 不涉及       | Kafka实例  | 1分钟        |
| group_msgs               | 消息堆积数 | 该指标用于统计Kafka实例中所有消费组中总堆积消息数。             | 0~1000000000 | Count  | 不涉及       | Kafka实例  | 1分钟        |
| instance_bytes_in_rate   | 生产流量  | 统计实例中每秒生产的字节数。<br>部分存量实例不支持此监控，具体以控制台为准。 | 0~1000000    | Byte/s | 1024(IEC) | Kafka实例  | 1分钟        |
| instance_bytes_out_rate  | 消费流量  | 统计实例中每秒消费的字节数。<br>部分存量实例不支持此监控，具体以控制台为准。 | 0~1000000    | Byte/s | 1024(IEC) | Kafka实例  | 1分钟        |
| current_partitions_usage | 分区使用率 | 该指标用于统计分区使用率。<br>部分存量实例不支持此监控，具体以控制台为准。  | 0~100        | %      | 不涉及       | Kafka实例  | 1分钟        |

节点监控指标

Kafka实例开启Smart Connect后，会创建两个或以上用于Smart Connect的代理。在“节点”页签中，“节点类型”选择“connector”，查看Smart Connect代理的相关监控。“节点类型”选择“broker”，查看Kafka实例代理的相关监控。

Smart Connect代理的监控指标包括：磁盘容量使用率、内存使用率、JVM堆内存使用率、节点存活状态和连接数。

表 14-5 节点支持的监控指标

| 指标ID                    | 指标名称       | 指标含义                             | 取值范围         | 单位      | 进制        | 测量对象（维度）  | 监控周期（原始指标） |
|-------------------------|------------|----------------------------------|--------------|---------|-----------|-----------|------------|
| broker_data_size        | 节点数据容量     | 该指标用于统计节点当前的消息数据大小。              | 0~5000000000 | Byte    | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_messages_in_rate | 消息生产速率     | 该指标用于统计每秒生产的消息数量。                | 0~500000     | Count/s | 不涉及       | Kafka实例节点 | 1分钟        |
| broker_bytes_out_rate   | 消费流量       | 该指标用于统计每秒消费的字节数。                 | 0~500000000  | Byte/s  | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_bytes_in_rate    | 生产流量       | 该指标用于统计每秒生产的字节数。                 | 0~500000000  | Byte/s  | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_fetch_mean       | 消费请求平均处理时长 | 统计Broker节点处理消费请求平均时长。            | 0~10000      | ms      | 不涉及       | Kafka实例节点 | 1分钟        |
| broker_produce_mean     | 生产请求平均处理时长 | 统计Broker节点处理生产请求平均时长。            | 0~10000      | ms      | 不涉及       | Kafka实例节点 | 1分钟        |
| broker_cpu_core_load    | CPU核均负载    | 该指标为从Kafka节点虚拟机层面采集的CPU每个核的平均负载。 | 0~20         | 不涉及     | 不涉及       | Kafka实例节点 | 1分钟        |

| 指标ID                   | 指标名称      | 指标含义                                                    | 取值范围                                                                 | 单位    | 进制  | 测量对象（维度）  | 监控周期（原始指标） |
|------------------------|-----------|---------------------------------------------------------|----------------------------------------------------------------------|-------|-----|-----------|------------|
| broker_disk_usage      | 磁盘容量使用率   | 该指标为从Kafka节点虚拟机层面采集的磁盘容量使用率。                            | 0~100                                                                | %     | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_memory_usage    | 内存使用率     | 该指标为Kafka节点虚拟机层面采集的内存使用率。                               | 0~100                                                                | %     | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_heap_usage      | JVM堆内存使用率 | 该指标从Kafka节点Kafka进程JVM中采集的堆内存使用率。                        | 0~100                                                                | %     | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_alive           | 节点存活状态    | 表示Kafka节点是否存活。<br>2020年4月及以后购买的实例，支持此监控项。               | <ul style="list-style-type: none"> <li>1：存活</li> <li>0：离线</li> </ul> | 不涉及   | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_connections     | 连接数       | 统计Kafka节点当前所有TCP连接数量。<br>2020年4月及以后购买的实例，支持此监控项。        | 0~65535                                                              | Count | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_cpu_usage       | CPU使用率    | 统计Kafka节点虚拟机的CPU使用率。<br>2020年4月及以后购买的实例，支持此监控项。         | 0~100                                                                | %     | 不涉及 | Kafka实例节点 | 1分钟        |
| broker_disk_read_await | 磁盘平均读操作耗时 | 该指标用于统计磁盘在测量周期内平均每个读IO的操作时长。<br>2020年6月及以后购买的实例，支持此监控项。 | >0                                                                   | ms    | 不涉及 | Kafka实例节点 | 1分钟        |

| 指标ID                        | 指标名称      | 指标含义                                                        | 取值范围        | 单位     | 进制        | 测量对象（维度）  | 监控周期（原始指标） |
|-----------------------------|-----------|-------------------------------------------------------------|-------------|--------|-----------|-----------|------------|
| broker_disk_write_await     | 磁盘平均写操作耗时 | 该指标用于统计磁盘在测量周期内平均每个写IO的操作时长。<br><br>2020年6月及以后购买的实例，支持此监控项。 | >0          | ms     | 不涉及       | Kafka实例节点 | 1分钟        |
| broker_total_bytes_in_rate  | 网络入流量     | 统计Broker节点每秒网络访问流入流量。<br><br>2020年6月及以后购买的实例，支持此监控项。        | 0~100000000 | Byte/s | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_total_bytes_out_rate | 网络出流量     | 统计Broker节点每秒网络访问流出流量。<br><br>2020年6月及以后购买的实例，支持此监控项。        | 0~100000000 | Byte/s | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_disk_read_rate       | 磁盘读流量     | 磁盘读操作流量。<br><br>2022年5月16号及以后购买的实例，支持此监控项。                  | >=0         | Byte/s | 1024(IEC) | Kafka实例节点 | 1分钟        |
| broker_disk_write_rate      | 磁盘写流量     | 磁盘写操作流量。<br><br>2022年5月16号及以后购买的实例，支持此监控项。                  | >=0         | Byte/s | 1024(IEC) | Kafka实例节点 | 1分钟        |

| 指标ID                    | 指标名称    | 指标含义                                                                                                                                                                | 取值范围  | 单位 | 进制  | 测量对象（维度）  | 监控周期（原始指标） |
|-------------------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|----|-----|-----------|------------|
| network_bandwidth_usage | 网络带宽利用率 | 网络带宽利用率。<br>仅部分实例支持此监控项，具体如下： <ul style="list-style-type: none"><li>2023年7月9号及以后购买的实例，支持此监控项。</li><li>2023年7月9号前购买的实例，如果在2023年7月9号及以后扩容代理数量，新扩容的代理支持此监控项。</li></ul> | 0~100 | %  | 不涉及 | Kafka实例节点 | 1分钟        |

主题监控指标

表 14-6 主题支持的监控指标

| 指标ID                | 指标名称 | 指标含义                                                | 取值范围        | 单位     | 进制        | 测量对象（维度）  | 监控周期（原始指标） |
|---------------------|------|-----------------------------------------------------|-------------|--------|-----------|-----------|------------|
| topic_bytes_in_rate | 生产流量 | 该指标用于统计每秒生产的字节数。<br>在“主题”页签，当“监控类型”为“基本监控”时，才包含该指标。 | 0~500000000 | Byte/s | 1024(IEC) | Kafka实例队列 | 1分钟        |

| 指标ID                   | 指标名称   | 指标含义                                                   | 取值范围           | 单位      | 进制        | 测量对象（维度）  | 监控周期（原始指标） |
|------------------------|--------|--------------------------------------------------------|----------------|---------|-----------|-----------|------------|
| topic_bytes_out_rate   | 消费流量   | 该指标用于统计每秒消费的字节数。<br>在“主题”页签，当“监控类型”为“基本监控”时，才包含该指标。    | 0~500000000    | Byte/s  | 1024(IEC) | Kafka实例队列 | 1分钟        |
| topic_data_size        | 队列数据容量 | 该指标用于统计队列当前的消息数据大小。<br>在“主题”页签，当“监控类型”为“基本监控”时，才包含该指标。 | 0~500000000000 | Byte    | 1024(IEC) | Kafka实例队列 | 1分钟        |
| topic_messages         | 队列消息总数 | 该指标用于统计队列当前的消息总数。<br>在“主题”页签，当“监控类型”为“基本监控”时，才包含该指标。   | ≥ 0            | Count   | 不涉及       | Kafka实例队列 | 1分钟        |
| topic_messages_in_rate | 消息生产速率 | 该指标用于统计每秒生产的消息数量。<br>在“主题”页签，当“监控类型”为“基本监控”时，才包含该指标。   | 0~500000       | Count/s | 不涉及       | Kafka实例队列 | 1分钟        |

| 指标 ID                      | 指标名称    | 指标含义                                                                                | 取值范围 | 单位      | 进制  | 测量对象（维度）  | 监控周期（原始指标） |
|----------------------------|---------|-------------------------------------------------------------------------------------|------|---------|-----|-----------|------------|
| partition_messages         | 分区消息数   | 该指标用于统计分区中当前的消息个数。<br>在“主题”页签，当“监控类型”为“分区监控”时，才包含该指标。                               | ≥ 0  | Count   | 不涉及 | Kafka实例队列 | 1分钟        |
| produced_messages          | 生产消息数   | 该指标用于统计目前生产的消息总数。<br>在“主题”页签，当“监控类型”为“分区监控”时，才包含该指标。                                | ≥ 0  | Count   | 不涉及 | Kafka实例队列 | 1分钟        |
| partition_messages_in_rate | 分区消息生产率 | 该指标用于统计分区每秒生产的消息数量。<br>在“主题”页签，当“监控类型”为“分区监控”时，才包含该指标。<br><br>部分存量实例不支持此监控，请联系客服解决。 | ≥ 0  | Count/s | 不涉及 | Kafka实例队列 | 1分钟        |

消费组监控指标

表 14-7 消费组支持的监控指标

| 指标 ID                    | 指标名称     | 指标含义                                                                          | 取值范围                 | 单位    | 进制  | 测量对象（维度）     | 监控周期（原始指标） |
|--------------------------|----------|-------------------------------------------------------------------------------|----------------------|-------|-----|--------------|------------|
| messages_consumed        | 分区已消费消息数 | 该指标用于统计当前消费组已经消费的消息个数。<br>在“消费组”页签，“主题”为指定的 Topic 名称，“监控类型”为“分区监控”时，才包含该指标。   | ≥ 0                  | Count | 不涉及 | Kafka 实例的消费组 | 1 分钟       |
| messages_remaining       | 分区可消费消息数 | 该指标用于统计消费组可消费的消息个数。<br>在“消费组”页签，“主题”为指定的 Topic 名称，“监控类型”为“分区监控”时，才包含该指标。      | ≥ 0                  | Count | 不涉及 | Kafka 实例的消费组 | 1 分钟       |
| topic_messages_remaining | 队列可消费消息数 | 该指标用于统计消费组指定队列可以消费的消息个数。<br>在“消费组”页签，“主题”为指定的 Topic 名称，“监控类型”为“基本监控”时，才包含该指标。 | 0~<br>( $2^{63}-1$ ) | Count | 不涉及 | Kafka 实例的消费组 | 1 分钟       |

| 指标ID                        | 指标名称             | 指标含义                                                                         | 取值范围                 | 单位    | 进制  | 测量对象（维度）    | 监控周期（原始指标） |
|-----------------------------|------------------|------------------------------------------------------------------------------|----------------------|-------|-----|-------------|------------|
| topic_messages_consumed     | 队列已消费消息数         | 该指标用于统计消费组指定队列当前已经消费的消息数。<br>在“消费组”页签，“主题”为指定的Topic名称，“监控类型”为“基本监控”时，才包含该指标。 | 0~<br>( $2^{63}-1$ ) | Count | 不涉及 | Kafka实例的消费组 | 1分钟        |
| consumer_messages_remaining | 消息堆积数（消费组可消费消息数） | 该指标用于统计消费组剩余可以消费的消息个数。<br>在“消费组”页签，“主题”为“全部队列”时，才包含该指标。                      | 0~<br>( $2^{63}-1$ ) | Count | 不涉及 | Kafka实例的消费组 | 1分钟        |
| consumer_messages_consumed  | 消费组已消费消息数        | 该指标用于统计消费组当前已经消费的消息数。<br>在“消费组”页签，“主题”为“全部队列”时，才包含该指标。                       | 0~<br>( $2^{63}-1$ ) | Count | 不涉及 | Kafka实例的消费组 | 1分钟        |

| 指标ID                                                           | 指标名称                | 指标含义                                                                                             | 取值范围           | 单位            | 进制      | 测量对象（维度）            | 监控周期（原始指标） |
|----------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------|----------------|---------------|---------|---------------------|------------|
| messa<br>ges_c<br>onsu<br>med_<br>per_m<br>in                  | 分区<br>消费<br>速率      | 统计消费组指定队列分区每分钟的消费数。<br>在“消费组”页签，“主题”为指定的Topic名称，“监控类型”为“分区监控”时，才包含该指标。<br>部分存量实例不支持此监控，具体以控制台为准。 | 0~3000<br>0000 | Count/<br>min | 不涉<br>及 | Kafka实<br>例的消费<br>组 | 1分钟        |
| topic_<br>messa<br>ges_c<br>onsu<br>med_<br>per_m<br>in        | 队列<br>消费<br>速率      | 统计消费组指定队列每分钟的消费数。<br>在“消费组”页签，“主题”为指定的Topic名称，“监控类型”为“基本监控”时，才包含该指标。<br>部分存量实例不支持此监控，具体以控制台为准。   | 0~3000<br>0000 | Count/<br>min | 不涉<br>及 | Kafka实<br>例的消费<br>组 | 1分钟        |
| consu<br>mer_<br>messa<br>ges_c<br>onsu<br>med_<br>per_m<br>in | 消费<br>组消<br>费速<br>率 | 统计消费组每分钟的消费数。<br>在“消费组”页签，“主题”为“全部Topic”时，才包含该指标。<br>部分存量实例不支持此监控，具体以控制台为准。                      | 0~3000<br>0000 | Count/<br>min | 不涉<br>及 | Kafka实<br>例的消费<br>组 | 1分钟        |

Smart Connect 监控指标

仅集群实例支持Smart Connect监控指标。

表 14-8 Smart Connect 支持的监控指标

| 指标 ID                       | 指标名称          | 指标含义                  | 取值范围                                                                    | 单位    | 进制  | 测量对象（维度）                  | 监控周期（原始指标） |
|-----------------------------|---------------|-----------------------|-------------------------------------------------------------------------|-------|-----|---------------------------|------------|
| kafka_wait_synchronize_data | 待同步 Kafka 数据量 | Kafka任务的待同步数据量。       | ≥ 0                                                                     | Count | 不涉及 | Kafka实例的 Smart Connect 任务 | 1分钟        |
| kafka_sync_hronize_rate     | Kafka每分钟同步数据量 | Kafka任务每分钟同步的数据量。     | ≥ 0                                                                     | Count | 不涉及 | Kafka实例的 Smart Connect 任务 | 1分钟        |
| task_status                 | 任务状态          | 当前任务状态。               | <ul style="list-style-type: none"><li>0: 任务异常</li><li>1: 任务正常</li></ul> | 不涉及   | 不涉及 | Kafka实例的 Smart Connect 任务 | 1分钟        |
| message_delay               | 消息时延          | 消息到达源端的时间与到达目标端的时间之差。 | ≥ 0                                                                     | ms    | 不涉及 | Kafka实例的 Smart Connect 任务 | 1分钟        |

使用Smart Connect监控指标时，请注意如下几点：

- Kafka双向数据复制的Smart Connect任务在监控中会被拆分为2个任务，分别为“Smart Connect任务名\_source\_0”和“Smart Connect任务名\_source\_1”。
- 如果Topic中的消息在进行下一次数据同步前，已经全部老化，此时实际是没有待同步的Kafka数据，但是Kafka数据同步监控指标使用的是包含老化数据的offset值，“待同步Kafka数据量”会显示老化的消息数。

维度

| Key               | Value                                                          |
|-------------------|----------------------------------------------------------------|
| kafka_instance_id | Kafka实例ID，例如：6a0df74a-fxxx-497bfdda6c8e。                       |
| kafka_broker      | Kafka实例节点，节点编号从0开始递增。例如3个节点的实例，节点为：broker-0,broker-1,broker-2。 |

| Key                     | Value                                  |
|-------------------------|----------------------------------------|
| kafka_topics            | Topic名称，例如：topic-01。                   |
| kafka_partitions        | Topic的分区号，例如：0。                        |
| kafka_groups_partitions | 消费组消费的Topic分区号，例如：0。                   |
| kafka_groups_topics     | 消费组消费的Topic名称，例如：topic-01。             |
| kafka_groups            | Kafka实例的消费组名称，例如：group-01。             |
| connector_task          | Smart Connect任务名称，例如：smart-connect-01。 |

- 请注意以下对于维度层级的说明：
- “kafka\_partitions”为“kafka\_topics”的子维度。
  - “kafka\_groups\_topics”和“kafka\_groups\_partitions”为“kafka\_groups”的子维度。

### 14.3 配置 Kafka 监控告警

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

表 14-9 Kafka 实例的告警策略和处理建议

| 指标ID                | 指标名称    | 测量对象 | 告警策略                                | 指标说明                         | 告警处理建议                                                                 |
|---------------------|---------|------|-------------------------------------|------------------------------|------------------------------------------------------------------------|
| broker_disk_usage   | 磁盘容量使用率 | 节点   | 告警阈值：原始值>80%<br>连续触发次数：1<br>告警级别：紧急 | 该指标为从Kafka节点虚拟机层面采集的磁盘容量使用率。 | 出现该告警时，需要修改实例 <b>存储空间</b> 。具体操作，请参考 <a href="#">变更实例规格</a> 。           |
| broker_memory_usage | 内存使用率   | 节点   | 告警阈值：原始值>90%<br>连续触发次数：3<br>告警级别：紧急 | 该指标为Kafka节点虚拟机层面采集的内存使用率。    | 出现该告警时，需要修改实例 <b>基准带宽/代理个数</b> ，即扩节点。具体操作，请参考 <a href="#">变更实例规格</a> 。 |

| 指标ID                   | 指标名称     | 测量对象 | 告警策略                                                                                | 指标说明                         | 告警处理建议                                                                                                                |
|------------------------|----------|------|-------------------------------------------------------------------------------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| current_partitions     | 分区数      | 实例   | 告警阈值：原始值>分区数上限的90%，不同实例规格分区数上限不同，具体参考 <a href="#">产品规格</a> 。<br>连续触发次数：1<br>告警级别：重要 | 该指标用于统计Kafka实例中已经使用的分区数量。    | 出现该告警时，如果业务后续还需要新增Topic，则需要修改实例 <a href="#">基准带宽/代理个数</a> 或将业务拆分至多个实例。修改实例基准带宽/代理个数的具体操作，请参考 <a href="#">变更实例规格</a> 。 |
| broker_cpu_usage       | CPU使用率   | 节点   | 告警阈值：原始值>90%<br>连续触发次数：3<br>告警级别：重要                                                 | 统计Kafka节点虚拟机的CPU使用率。         | 出现该告警时，先检查该监控是否长期处于接近或超过告警阈值状态，如果是，需要修改实例 <a href="#">基准带宽/代理个数</a> ，即扩节点。具体操作，请参考 <a href="#">变更实例规格</a> 。           |
| group_msgs             | 堆积消息数    | 实例   | 告警阈值：原始值>积压上限的90%，积压上限由您根据业务实际情况设定<br>连续触发次数：1<br>告警级别：重要                           | 该指标用于统计Kafka实例中所有消费组中总堆积消息数。 | 出现该告警时，首先排查是否有闲置消费组，如果有，则删除。其次，可以考虑加快消费速度，例如增加组内消费者数量等。                                                               |
| topic_messages_remaind | 队列可消费消息数 | 消费组  | 告警阈值：原始值>积压上限的90%，积压上限由您根据业务实际情况设定<br>连续触发次数：1<br>告警级别：重要                           | 该指标用于统计消费组指定队列可以消费的消息个数。     | 出现该告警时，首先排查消费者代码逻辑是否有误，例如消费者出现了异常不再消费等。其次，可以考虑加快消息的消费，例如增加队列消费者，并确保分区数大于或等于消费者数。                                      |

| 指标ID        | 指标名称 | 测量对象            | 告警策略                              | 指标说明                          | 告警处理建议                                                                   |
|-------------|------|-----------------|-----------------------------------|-------------------------------|--------------------------------------------------------------------------|
| task_status | 任务状态 | Smart Connect任务 | 告警阈值：原始值<1<br>连续触发次数：3<br>告警级别：重要 | 该指标用于统计Smart Connect任务状态是否正常。 | 出现该告警时，首先在Smart Connect任务列表页面暂停任务，然后再重新启动任务，观察任务状态指标是否恢复。若仍未恢复，建议联系客服处理。 |

## 配置 Kafka 告警规则

以下步骤指导您配置指定Kafka实例的告警规则。

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 通过以下任意一种方法，进入监控页面。

- 在Kafka实例名称后，单击“查看监控数据”，进入云监控该实例的监控指标页面。
- 单击Kafka实例名称，进入实例详情页。在左侧导航栏单击“监控 > 监控详情”，进入监控页面。

**步骤4** 在实例监控指标页面中，找到需要创建告警的指标项，鼠标移动到指标区域，然后单击指标右上角的，跳转到创建告警规则页面。

**步骤5** 在告警规则页面，设置告警信息。

创建告警规则操作，请查看[创建告警规则](#)。

- 设置告警名称和告警的描述。
- 设置告警策略。

如下图所示，在进行指标监控时，如果连续1次，磁盘容量使用率原始值 $\geq 80\%$ ，则产生告警，如果未及时处理，则产生告警通知。

图 14-3 设置告警策略



- 设置“发送通知”开关。  
当开启时，设置通知方式和不同方式对应的通知参数。
- 单击“立即创建”，等待创建告警规则成功。

----结束

## 查看 Kafka 实例告警规则

以下步骤指导您查询指定Kafka实例的所有告警规则。

**步骤1** 登录[Kafka控制台](#)。

**步骤2** 在管理控制台左上角单击 ，选择Kafka实例所在的区域。

**步骤3** 在管理控制台左上角单击 ，选择“管理与监管 > 云监控服务 CES”，进入云监控服务总览页面。

**步骤4** 在左侧导航栏单击“云服务监控”，进入“云服务监控”页面。

**步骤5** 在顶部搜索框中输入“分布式消息服务”，按Enter。

**步骤6** 单击“分布式消息服务 DMS”，进入“资源详情”页签。

**步骤7** 在待查看告警规则的Kafka实例所在行，单击“更多 > 查看告警规则”，弹出“查看告警规则”对话框，查看此Kafka实例的所有告警规则。

----结束

# 15 查看 Kafka 审计日志

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

## 前提条件

已开通CTS。

## CTS 支持的 DMS for Kafka 操作

表 15-1 云审计服务支持的 DMS for Kafka 操作列表

| 操作名称            | 资源类型  | 事件名称                                   |
|-----------------|-------|----------------------------------------|
| 创建DMS实例订单成功     | kafka | createDMSInstanceOrderSuccess          |
| 创建DMS实例任务执行成功   | kafka | createDMSInstanceTaskSuccess           |
| 创建DMS实例订单失败     | kafka | createDMSInstanceOrderFailure          |
| 创建DMS实例任务执行失败   | kafka | createDMSInstanceTaskFailure           |
| 删除创建失败的DMS实例成功  | kafka | deleteDMSCreateFailureInstancesSuccess |
| 删除创建失败的DMS实例失败  | kafka | deleteDMSCreateFailureInstancesFailure |
| 删除DMS实例任务执行成功   | kafka | deleteDMSInstanceTaskSuccess           |
| 删除DMS实例任务执行失败   | kafka | deleteDMSInstanceTaskFailure           |
| 批量删除DMS实例任务     | kafka | batchDeleteDMSInstanceTask             |
| 提交批量删除DMS实例请求成功 | kafka | batchDeleteDMSInstanceSuccess          |

| 操作名称            | 资源类型     | 事件名称                               |
|-----------------|----------|------------------------------------|
| 批量删除DMS实例任务执行成功 | kafka    | batchDeleteDMSInstanceTaskSuccess  |
| 提交批量删除DMS实例请求失败 | kafka    | batchDeleteDMSInstanceFailure      |
| 批量删除DMS实例任务执行失败 | kafka    | batchDeleteDMSInstanceTaskFailure  |
| 提交修改DMS实例订单请求成功 | kafka    | modifyDMSInstanceOrderSuccess      |
| 提交修改DMS实例订单请求失败 | kafka    | modifyDMSInstanceOrderFailure      |
| 提交扩容实例请求成功      | kafka    | extendDMSInstanceSuccess           |
| 扩容DMS实例任务执行成功   | kafka    | extendDMSInstanceTaskSuccess       |
| 提交扩容实例请求失败      | kafka    | extendDMSInstanceFailure           |
| 扩容DMS实例任务执行失败   | kafka    | extendDMSInstanceTaskFailure       |
| 提交重置DMS实例密码请求成功 | kafka    | resetDMSInstancePasswordSuccess    |
| 提交重置DMS实例密码请求失败 | kafka    | resetDMSInstancePasswordFailure    |
| 提交重启DMS实例请求成功   | kafka    | restartDMSInstanceSuccess          |
| 重启DMS实例任务执行成功   | kafka    | restartDMSInstanceTaskSuccess      |
| 提交重启DMS实例请求失败   | kafka    | restartDMSInstanceFailure          |
| 重启DMS实例任务执行失败   | kafka    | restartDMSInstanceTaskFailure      |
| 提交批量重启DMS实例请求成功 | instance | batchRestartDMSInstanceSuccess     |
| 批量重启DMS实例任务执行成功 | kafka    | batchRestartDMSInstanceTaskSuccess |
| 提交批量重启DMS实例请求失败 | instance | batchRestartDMSInstanceFailure     |
| 批量重启DMS实例任务执行失败 | kafka    | batchRestartDMSInstanceTaskFailure |

| 操作名称                | 资源类型  | 事件名称                             |
|---------------------|-------|----------------------------------|
| 提交修改DMS实例信息请求成功     | kafka | modifyDMSInstanceInfoSuccess     |
| 修改DMS实例信息任务执行成功     | kafka | modifyDMSInstanceInfoTaskSuccess |
| 提交修改DMS实例信息请求失败     | kafka | modifyDMSInstanceInfoFailure     |
| 修改DMS实例信息任务执行失败     | kafka | modifyDMSInstanceInfoTaskFailure |
| 删除后台任务成功            | kafka | deleteDMSBackendJobSuccess       |
| 删除后台任务失败            | kafka | deleteDMSBackendJobFailure       |
| 开启Smart Connect成功   | kafka | CreateConnectorSuccess           |
| 开启Smart Connect失败   | kafka | createConnectorFailure           |
| 创建Smart Connect任务成功 | kafka | createConnectorTaskSuccess       |
| 创建Smart Connect任务失败 | kafka | createConnectorTaskFailure       |
| 修改Smart Connect任务成功 | kafka | putConnectorTaskSuccess          |
| 修改Smart Connect任务失败 | kafka | putConnectorTaskFailure          |
| 删除Smart Connect任务成功 | kafka | deleteConnectorTaskSuccess       |
| 删除Smart Connect任务失败 | kafka | deleteConnectorTaskFailure       |
| 重启Smart Connect任务成功 | kafka | restartConnectorSuccess          |
| 重启Smart Connect任务失败 | kafka | restartConnectorFailure          |
| 暂停Smart Connect任务成功 | kafka | pauseConnectorSuccess            |
| 暂停Smart Connect任务失败 | kafka | pauseConnectorFailure            |
| 启动Smart Connect任务成功 | kafka | resumeConnectorSuccess           |

| 操作名称                | 资源类型  | 事件名称                                 |
|---------------------|-------|--------------------------------------|
| 启动Smart Connect任务失败 | kafka | resumeConnectorFailure               |
| 检查连通性成功             | kafka | ValidateConnectorConnectivitySuccess |
| 检查连通性失败             | kafka | ValidateConnectorConnectivityFailure |
| 关闭Smart Connect成功   | kafka | DeleteConnectorSuccess               |
| 关闭Smart Connect失败   | kafka | DeleteConnectorFailure               |
| 创建Smart Connect节点成功 | kafka | createConnectorNodeTaskSuccess       |
| 创建Smart Connect节点失败 | kafka | createConnectorNodeTaskFailure       |
| 删除Smart Connect节点成功 | kafka | deleteConnectorNodeTaskSuccess       |
| 删除Smart Connect节点失败 | kafka | deleteConnectorNodeTaskFailure       |
| 冻结DMS实例任务执行成功       | kafka | freezeDMSInstanceTaskSuccess         |
| 冻结DMS实例任务执行失败       | kafka | freezeDMSInstanceTaskFailure         |
| 解冻DMS实例任务执行成功       | kafka | unfreezeDMSInstanceTaskSuccess       |
| 解冻DMS实例任务执行失败       | kafka | unfreezeDMSInstanceTaskFailure       |
| Kafka实例创建Topic成功    | kafka | Kafka_create_topicSuccess            |
| Kafka实例创建Topic失败    | kafka | Kafka_create_topicFailure            |
| Kafka实例删除Topic成功    | kafka | Kafka_delete_topicsSuccess           |
| Kafka实例删除Topic失败    | kafka | Kafka_delete_topicsFailure           |
| 开启自动创建Topic成功       | kafka | enable_auto_topicSuccess             |
| 开启自动创建Topic失败       | kafka | enable_auto_topicFailure             |

| 操作名称              | 资源类型  | 事件名称                               |
|-------------------|-------|------------------------------------|
| 修改Topic成功         | kafka | Kafka_alter_topicsSuccess          |
| 修改Topic失败         | kafka | Kafka_alter_topicsFailure          |
| 执行Topic分区平衡任务成功   | kafka | kafka_reassignmentTaskSuccess      |
| 执行Topic分区平衡任务失败   | kafka | kafka_reassignmentTaskFailure      |
| 提交执行Topic分区平衡请求成功 | kafka | kafka_reassignmentSuccess          |
| 提交执行Topic分区平衡请求失败 | kafka | kafka_reassignmentFailure          |
| 重置消费组偏移量成功        | kafka | Kafka_reset_consumer_offsetSuccess |
| 重置消费组偏移量失败        | kafka | Kafka_reset_consumer_offsetFailure |
| 批量删除消费组成功         | kafka | Kafka_batch_delete_groupSuccess    |
| 批量删除消费组失败         | kafka | Kafka_batch_delete_groupFailure    |
| 创建用户成功            | kafka | createUserSuccess                  |
| 创建用户失败            | kafka | createUserFailure                  |
| 删除用户成功            | kafka | deleteUserSuccess                  |
| 删除用户失败            | kafka | deleteUserFailure                  |
| 更新用户策略成功          | kafka | updateUserPoliciesTaskSuccess      |
| 更新用户策略失败          | kafka | updateUserPoliciesTaskFailure      |
| 回收站恢复实例成功         | kafka | out_recycleTaskSuccess             |
| 回收站恢复实例失败         | kafka | out_recycleTaskFailure             |

查看审计日志

查看DMS for Kafka云审计日志，请参考[查看审计事件](#)。