

分布式消息服务 Kafka 版

故障排除

文档版本 01
发布日期 2025-09-16



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 连接问题排查和解决.....	1
2 消息从生产到消费时延高达 6 分钟.....	4
3 生产消息失败.....	6
4 删除 Topic 失败.....	7
5 在 Windows 服务器中无法登录 Kafka Manager.....	8
6 生产或消费消息时，报 Topic {{topic_name}} not present in metadata after 60000 ms 错误.....	10
7 Flink 1.15 消费进度提交失败.....	11

1 Kafka 连接问题排查和解决

概述

本章节主要描述Kafka连接问题的排查和解决。

问题分类

当您发现与Kafka实例连接出现异常时，可以根据本文的内容，从以下几个方面进行排查。

- [检查网络](#)
- [检查Consumer和Producer的配置](#)
- [客户端连接问题-Java客户端](#)
- [客户端连接问题-Go客户端](#)

检查网络

在连接Kafka实例前，需要确保客户端与Kafka实例之间可以相互连接，如果无法连接，建议先检查网络连接是否正常。

已开启sasL_ssl访问的Kafka实例为例，执行如下命令：

```
curl -kv {ip}:{port}
```

- 网络正常，如下：

```
[root@ecs-5d2f ~]# curl -kv 192.168.0.52:9093
* Rebuilt URL to: 192.168.0.52:9093/
* Trying 192.168.0.52...
* TCP_NODELAY set
* Connected to 192.168.0.52 (192.168.0.52) port 9093 (#0)
> GET / HTTP/1.1
> Host: 192.168.0.52:9093
> User-Agent: curl/7.61.1
> Accept: */*
>
Warning: Binary output can mess up your terminal. Use "--output -" to tell
Warning: curl to output it to your terminal anyway, or consider "--output
Warning: <FILE>" to save to a file.
* Failed writing body (0 != 7)
* Closing connection 0
```

- 网络异常/不通，如下：

```
[root@ecs-5d2f ~]# curl -kv 192.168.0.52:9093
* Rebuilt URL to: 192.168.0.52:9093/
* Trying 192.168.0.52...
* TCP_NODELAY set
* connect to 192.168.0.52 port 9093 failed: Connection timed out
* Failed to connect to 192.168.0.52 port 9093: Connection timed out
* Closing connection 0
curl: (7) Failed to connect to 192.168.0.52 port 9093: Connection timed out
```

解决方案：

1. 检查是否同一个VPC。如果不是，请参考[跨VPC访问](#)。
2. 检查安全组规则是否配置正确。请参考[如何选择和配置安全组](#)。

检查 Consumer 和 Producer 的配置

查看日志，检查Consumer和Producer初始化时打印的参数是否符合预期，是否和自己在配置文件中设置的一致。

如果不一致，请检查配置文件的写法。

Java 客户端常见报错

- 未关闭域名校验

出现下图的错误信息：

```
Caused by: java.lang.Thread.run(Thread.java:748)
Caused by: javax.net.ssl.SSLHandshakeException: General SSLEngine problem
    at sun.security.ssl.Alerts.getSSLException(Alerts.java:192)
    at sun.security.ssl.SSLEngineImpl.fatal(SSLEngineImpl.java:1789)
    at sun.security.ssl.Handshaker.fatalSE(Handshaker.java:318)
    at sun.security.ssl.Handshaker.fatalSE(Handshaker.java:318)
    at sun.security.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:1639)
    at sun.security.ssl.ClientHandshaker.processMessage(ClientHandshaker.java:223)
    at sun.security.ssl.Handshaker.processLoop(Handshaker.java:1037)
    at sun.security.ssl.Handshaker$1.run(Handshaker.java:978)
    at sun.security.ssl.Handshaker$1.run(Handshaker.java:967)
    at java.security.AccessController.doPrivileged(Native Method)
    at sun.security.ssl.Handshaker$DelegatedTask.run(Handshaker.java:1459)
    at org.apache.kafka.common.network.SslTransportLayer.runDelegatedTasks(SslTransportLayer.java:482)
    at org.apache.kafka.common.network.SslTransportLayer.handshakeUnwrap(SslTransportLayer.java:484)
    at org.apache.kafka.common.network.SslTransportLayer.doHandshake(SslTransportLayer.java:348)
    ... 7 more
Caused by: java.security.cert.CertificateException: No subject alternative names matching IP address 10.166.37.165 found
    at sun.security.util.HostnameChecker.matchIP(HostnameChecker.java:168)
    at sun.security.util.HostnameChecker.match(HostnameChecker.java:94)
    at sun.security.ssl.X509TrustManagerImpl.checkIdentity(X509TrustManagerImpl.java:462)
    at sun.security.ssl.X509TrustManagerImpl.checkIdentity(X509TrustManagerImpl.java:442)
    at sun.security.ssl.X509TrustManagerImpl.checkTrusted(X509TrustManagerImpl.java:261)
    at sun.security.ssl.X509TrustManagerImpl.checkServerTrusted(X509TrustManagerImpl.java:144)
    at sun.security.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:1626)
    ... 16 more
(Kafka.admin.TopicCommand$)
```

解决方案：检查consumer.properties和producer.properties文件，ssl.endpoint.identification.algorithm参数必须设置为空，关闭域名校验开关。

```
ssl.endpoint.identification.algorithm=
```

- 加载SSL证书文件失败

出现下图的错误信息：

```
[2020-09-28T06:35:38.654][ERROR][logstash.outputs.kafka] Unable to create Kafka producer from given configuration [-kafka_error_message=org.apache.kafka.common.KafkaException: Failed to construct kafka producer, :cause=org.apache.kafka.common.KafkaException: org.apache.kafka.common.KafkaException: Failed to load SSL keystore /opt/cloud/logstash/pipeline/001/logstash.cn.north.4/client.truststore.jks of type PKCS]
```

解决方案：

- a. 请检查对应地址的client.jks文件是否存在。
- b. 检查进程和文件所属权限。
- c. 检查consumer.properties和producer.properties文件中的ssl.truststore.password参数是否配置正确。

ssl.truststore.password为服务器证书密码，不可更改，需要保持为**dms@kafka**

```
ssl.truststore.password=dms@kafka
```

- Topic名称错误

出现下图的错误信息：

```
020-05-11 01:11:23,584 INFO [eventpull-thread0] [impl.KafkaClientImpl:207] .....ready poll ,topic is CSBPromotionManageService_PromotionTopic
020-05-11 01:11:23,704 INFO [eventpull-thread0] [kafka.PullToDatal:71] pull event from kafka cost time 200, topic CSBPromotionManageService_PromotionTopic, eventList []
020-05-11 01:11:24,620 ERROR [PublicEventToKafka-Thread] [impl.KafkaClientImpl:208] send event to kafka failed, topic=[CSBPromotionCouponService_CouponTopic], eventId = [01700-99999]
CommonsMaven3-nl ex...
[2020-05-11 01:11:24,620] ERROR [PublicEventToKafka-Thread] [impl.KafkaClientImpl:208] send event to kafka failed, topic=[CSBPromotionCouponService_CouponTopic], eventId = [01700-99999]
[2020-05-11 01:11:24,717] INFO [pool-20-thread-1] [impl.KafkaClientImpl:100] ready.getTopicList
020-05-11 01:11:24,724 INFO [pool-20-thread-1] [impl.KafkaClientImpl:107] getTopicList cost time 6
020-05-11 01:11:24,724 INFO [pool-20-thread-1] [impl.KafkaClientImpl:112] end.getTopicList
020-05-11 01:11:24,803 INFO [eventpull-thread0] [impl.KafkaClientImpl:207] .....ready poll ,topic is CSBPromotionCouponService_CouponTopic
```

解决方案：用户重新创建Topic或者打开自动创建Topic开关。

Go 客户端常见报错

使用GO客户端无法连接Kafka SSL，报“first record does not look like a TLS handshake”错误。

解决方案：go默认是关闭的，2021年1月前创建的实例，需要打开TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256或TLS_RSA_WITH_AES_128_CBC_SHA256加密套件。2021年1月以及之后创建的实例，需要打开TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256加密套件。

2 消息从生产到消费时延高达 6 分钟

问题现象

消息从生产到消费的端到端时延偶现到达6分钟，业务对消息时延较为敏感。

分析过程

1. 怀疑是业务堆积，处理慢导致时延。
通过查看监控数据，发现业务请求并不是很多，堆积最多时也只有50条消息，同时每秒新增消息只有10条左右，应该还没有到达处理极限。
2. 排查EIP流量情况，发现入流量存在下降的场景。
联系EIP服务技术人员协助排查，未发现问题。
3. 从业务日志中分析消费组行为。
通过查看服务端日志，消费组存在大量rebalance动作，大部分rebalance都会秒级完成，但偶尔会有分钟级别的rebalance耗时，而rebalance过程中是无法正常消费的，只有在rebalance动作完成才可以进行消费。
该现象与问题现象描述的偶现长时间时延行为相吻合，问题确定。

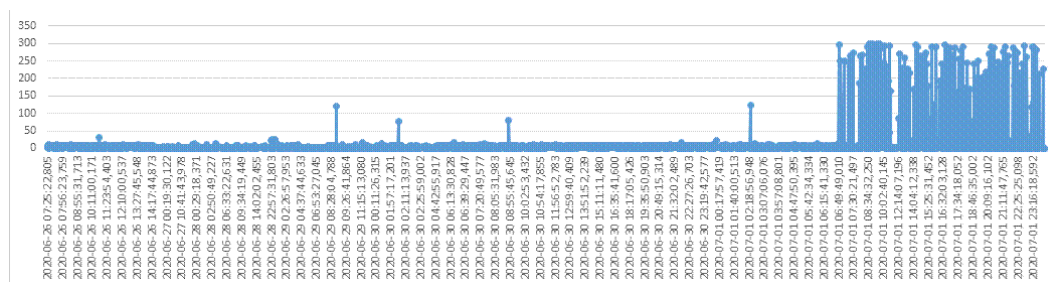
详细分析

查看用户消费组行为日志文件，文件中存在以下三种日志：

- Preparing to rebalance group 1
表示消费组开始进行rebalance，消费组状态变更为REBALANCING。
- Stabilized group
表示消费组rebalance完成，消费组状态变更为STABILIZED。
- Member consumer-xxx in group 1 has failed
表示消费组内的某一个消费者超时退出，一般触发场景是**消费者长时间没有与服务端通信导致**，比如处理消息耗时太久阻塞等。

对文件中每次Preparing到Stabilized完成的时间进行计算得到以下结果图。图中时间为UTC+0时间，对应北京时间需要加8小时。

图 2-1 消费组 rebalance 图



从以上图中可以看出，消费组rebalance的性能在7月1号06:49（即北京时间7月1号14:49）以后存在明显劣化的情况，导致客户端异常。

根因描述

用户业务中偶尔会存在某一个消费者无法及时响应rebalance的动作，导致整个消费组消费阻塞，一直到该消费者响应rebalance动作为止。

问题规避

1. 建议用户根据业务区分不同消费组使用，降低单消费者阻塞导致的影响访问。
2. `max.poll.interval.ms`用于设置消费组请求消费消息的最大时间间隔，如果消费者在超时前没有发起下一次消费请求，服务端会触发rebalance。调大`max.poll.interval.ms`的默认配置，降低问题频率。

问题解决措施

1. 区分业务使用不同消费组。
2. 需要客户侧排查自身业务，优化自身业务处理逻辑，提高处理效率，降低阻塞时间。

背景知识介绍

消费组可以简单认为有两种状态REBALANCING和STABILIZED。

- REBALANCING：消费组元数据发生变化，该状态下消费组中的所有消费者都**无法进行正常的业务消费**，该场景触发场景为消费组内有新的消费者加入或有已经建立连接的消费者退出。
- STABILIZED：rebalance完成，消费组处于稳定状态，该状态下消费组中的消费者**可以进行正常的业务消费**，触发条件是，当前消费组内的所有消费者都同步完成新的消费组元数据，包括之前已经同步过的消费者，也需要重新同步。

消费组简单流程如下：

1. 有新的消费者加入或退出，服务端记录的消费组元数据更新，服务端更新消费组进入REBALANCING状态。
2. 服务端等待所有消费者（包含已有的消费者）同步最新的元数据。
3. 所有消费者同步完最新的元数据后，服务端更新消费组状态为STABILIZED。
4. 消费者开始正常的消费业务。

3 生产消息失败

问题现象

系统提示 “Disk error when trying to access log file on the disk” 错误。

根因描述

节点磁盘使用率超高导致生产消息失败。

问题解决措施

扩容实例的磁盘空间，具体操作请参考[变更Kafka实例规格](#)。

4 删除 Topic 失败

问题现象

删除Topic不生效，删除后该Topic仍然存在。

根因描述

实例开启了自动创建Topic功能，且有消费者正在连接该Topic。如果没有停止业务，删除Topic后，还会有消息生产行为，并自动创建Topic。

问题解决措施

关闭Kafka自动创建Topic功能，才可以正常删除Topic。关闭方法请参考[配置Kafka自动创建Topic](#)。

5 在 Windows 服务器中无法登录 Kafka Manager

问题现象

在Windows服务器的浏览器中输入Kafka Manager地址，无法登录Kafka Manager，报错信息如下图。



根因描述

1. Windows服务器和Kafka实例不在同一个VPC同一个子网内，或者安全组配置不正确，导致网络无法连通。
2. Kafka Manager异常。

问题解决措施

1. 检查Windows服务器和Kafka实例是否在同一个VPC同一个子网内？
 - 是，执行2。
 - 否，修改Windows服务器的VPC和子网，与Kafka实例一致。

2. 检查安全组是否配置正确？安全组的配置请参考[选择和配置安全组](#)。
 - 是，执行[3](#)。
 - 否，修改安全组配置。
3. 在Kafka实例控制台，重启Kafka Manager，具体操作请参考[重启Kafka Manager](#)。

6 生产或消费消息时，报 Topic {{topic_name}} not present in metadata after 60000 ms 错误

问题现象

多可用区的Kafka实例，在某个可用区故障后，Kafka客户端在生产或消费消息时，可能会报Topic {{topic_name}} not present in metadata after 60000 ms的错误，如下图所示。

```
ssl.secure.random.implementation = null
ssl.trustmanager.algorithm = PKIX
ssl.truststore.location = null
ssl.truststore.password = null
ssl.truststore.type = JKS
transaction.timeout.ms = 60000
transactional.id = null
value.serializer = class org.apache.kafka.common.serialization.StringSerializer
(org.apache.kafka.clients.producer.ProducerConfig)
[2021-10-29 15:44:44,141] INFO Kafka version: 2.3.0 (org.apache.kafka.common.utils.AppInfoParser)
[2021-10-29 15:44:44,141] INFO Kafka commitId: fclaa116b661c8a (org.apache.kafka.common.utils.AppInfoParser)
[2021-10-29 15:44:44,141] INFO Kafka startTimeMs: 1635493484139 (org.apache.kafka.common.utils.AppInfoParser)
[2021-10-29 15:45:44,146] ERROR produce message failed. error msg: Topic topic-test not present in metadata after 60000 ms. (org.example.Producer)
[2021-10-29 15:46:44,247] ERROR produce message failed. error msg: Topic topic-test not present in metadata after 60000 ms. (org.example.Producer)
[2021-10-29 15:46:51,418] WARN [Producer clientId=producer-1] Connection to node -3 (/100.85.120.91:9094) could not be established. Broker may not be available. (org.apache.kafka.clients.NetworkClient)
[2021-10-29 15:46:51,684] INFO [Producer clientId=producer-1] Cluster ID: t0R4RgFHTN2pjUhiJqkFPQ (org.apache.kafka.clients.Metadata)
[2021-10-29 15:46:51,733] INFO produce message success. partition: 1, offset: 9335 (org.example.Producer)
[2021-10-29 15:46:51,809] INFO produce message success. partition: 4, offset: 9336 (org.example.Producer)
[2021-10-29 15:46:51,920] INFO produce message success. partition: 5, offset: 9335 (org.example.Producer)
[2021-10-29 15:46:52,005] INFO produce message success. partition: 2, offset: 9336 (org.example.Producer)
[2021-10-29 15:46:52,112] INFO produce message success. partition: 3, offset: 9327 (org.example.Producer)
[2021-10-29 15:46:52,206] INFO produce message success. partition: 8, offset: 9324 (org.example.Producer)
[2021-10-29 15:46:52,308] INFO produce message success. partition: 9, offset: 9332 (org.example.Producer)
[2021-10-29 15:46:52,410] INFO produce message success. partition: 6, offset: 9332 (org.example.Producer)
[2021-10-29 15:46:52,509] INFO produce message success. partition: 7, offset: 9335 (org.example.Producer)
[2021-10-29 15:46:52,608] INFO produce message success. partition: 0, offset: 9335 (org.example.Producer)
[2021-10-29 15:46:52,709] INFO produce message success. partition: 1, offset: 9336 (org.example.Producer)
[2021-10-29 15:46:52,808] INFO produce message success. partition: 4, offset: 9337 (org.example.Producer)
```

问题解决措施

以下三种措施都可以解决此问题，请根据实际情况任意选择一种。

- 升级Kafka客户端的版本到2.7或以上版本，并设置“socket.connection.setup.timeout.ms”大于1s，且小于“request.timeout.ms/Kafka服务端节点数”。
- 修改Kafka客户端的“request.timeout.ms”大于“127s”。
- 修改Kafka客户端Linux系统的网络参数“net.ipv4.tcp_syn_retries”为“3”。

7 Flink 1.15 消费进度提交失败

问题现象

Flink 1.15版本消费Kafka时，消费进度提交失败，持续报错：
COORDINATOR_NOT_AVAILABLE。

根因描述

Flink 1.15版本使用的Kafka客户端存在bug，当某次消费进度提交失败后，客户端将coordinator置为不可用且无法自动恢复。

问题解决措施

- 重启Flink job。
- 升级Flink至1.16及以上版本。