数据传输服务

API 文档

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2025 腾讯云版权所有

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

【商标声明】

腾讯云

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

【服务声明】

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

【联系我们】

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

文档目录

API 文档
- 产品版本
- 更新历史
- 简介
- API 概览
- 调用方式
  - 请求结构
  - 公共参数
  - 签名方法 v3
  - 签名方法
  - 返回结果
  - 参数类型
- （新版）数据迁移相关接口
  - 查询数据迁移任务列表
  - 购买迁移任务
  - 配置迁移服务
  - 校验迁移任务
  - 查询迁移校验任务结果
  - 跳过迁移校验检查项
  - 启动数据迁移任务
  - 完成数据迁移任务
  - 恢复暂停中的迁移任务
  - 查询可迁移的实例列表
  - 查询某个迁移任务详情
  - 下线数据迁移任务
  - 隔离数据迁移任务
  - 调整实例规格
  - 修改迁移名称
  - 修改任务运行时属性
  - 暂停迁移任务
  - 解除隔离数据迁移任务
  - 重试迁移任务
  - 终止数据迁移任务
  - 修改迁移任务的传输速率
- 数据同步相关接口
  - 查询同步任务信息
  - 创建同步任务
  - 配置同步任务
  - 校验同步任务
  - 查询同步校验任务结果
  - 跳过同步校验检查项
  - 启动同步任务
  - 隔离同步任务
  - 解除隔离同步任务
  - 暂停同步任务
  - 恢复被暂停的同步任务
  - 重试同步任务
  - 结束同步任务
  - 下线同步任务
  - 调整同步任务规格
  - 修改同步任务配置
  - 创建修改同步配置的校验任务
  - 查询修改对象的校验任务的结果
  - 开始修改配置流程
  - 修改同步任务的传输速率

数据一致性校验相关接口
查询一致性校验任务详情
终止一致性校验任务
删除一致性校验任务
修改一致性校验任务
修改一致性校验任务名称
创建一致性校验任务
启动一致性校验任务
查询一致性校验任务列表
（新版）数据订阅相关接口
创建订阅任务
配置订阅任务
校验订阅任务
查询订阅校验结果
启动订阅任务
修改订阅对象
查询订阅任务详情
获取数据订阅实例列表
重试订阅任务
创建消费组
获取数据订阅的消费组详情
修改消费组备注
修改消费组密码
查询kafka offset
重置kafka offset
删除消费组
重置订阅任务
修改订阅实例自动续费标识
修改数据订阅实例的名称
查询订阅实例是否可以退换
销毁订阅任务
下线已隔离的数据订阅任务
数据结构
错误码
数据传输服务 API 2018-03-30
产品版本
更新历史
简介
API 概览
调用方式
请求结构
公共参数
签名方法 v3
签名方法
返回结果
参数类型
数据传输服务相关接口
查询数据迁移任务
撤销数据迁移任务
启动数据迁移任务
修改数据迁移任务
获取迁移校验结果
删除数据迁移任务
创建数据迁移任务
创建校验迁移任务
完成数据迁移任务
数据订阅相关接口

修改订阅实例自动续费标识
重置数据订阅实例
下线已隔离的数据订阅实例
修改数据订阅实例的IP和端口号
修改数据订阅通道的订阅规则
修改数据订阅实例的名称
修改数据订阅实例通道的消费时间点
隔离订阅实例
获取数据订阅实例列表
查询订阅实例配置
查询任务执行结果
创建订阅对象
配置数据订阅通道
数据结构
错误码

API 文档
产品版本

最近更新时间：2025-06-20 01:31:42

您正在浏览dts产品文档的版本: 2021-12-06

最新版本

- 2021-12-06 (当前版本)

以往版本

- 2018-03-30

更新历史

最近更新时间：2025-06-24 01:22:37

第 41 次发布

发布时间：2025-06-24 01:22:28

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DBEndpointInfo
  - 新增成员：CcnOwnerUin

第 40 次发布

发布时间：2025-04-29 01:31:11

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ModifyConsumerGroupPassword
  - **删除入参：** OldPassword

第 39 次发布

发布时间：2025-04-02 16:33:03

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DBInfo
  - 新增成员：EncryptConn
- DynamicOptions
  - 新增成员：KafkaOption, FilterBeginCommit, FilterCheckpoint, DealOfExistSameTable, StartPosition

第 38 次发布

发布时间：2025-01-09 01:16:34

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeSubscribeJobs
  - 新增入参：SubscribeIds
- DescribeSyncJobs
  - 新增入参：JobIds, SrcInfoPattern, DstInfoPattern

第 37 次发布

发布时间：2025-01-03 01:17:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateSyncJob
  - 新增入参：TimeSpan

第 36 次发布

发布时间：2024-12-27 11:10:08

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SyncJobInfo
  - 新增成员：OptObjStatus

第 35 次发布

发布时间：2024-12-26 01:31:15

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ModifyConsumerGroupPassword
  - **修改入参：** OldPassword

第 34 次发布

发布时间：2024-12-25 01:31:41

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SubscribeCheckStepTip
  - **修改成员：** Message, HelpDoc

第 33 次发布

发布时间：2024-12-20 01:25:52

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Options
  - 新增成员：StartPosition

第 32 次发布

发布时间：2024-12-03 01:17:27

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SubsErr
  - 新增成员：Reason, Solution
  - **修改成员：** Message

第 31 次发布

发布时间：2024-07-24 01:36:02

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- StopCompare
  - 新增入参：ForceStop

第 30 次发布

发布时间：2024-07-03 01:15:22

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Endpoint
  - 新增成员：SetId

第 29 次发布

发布时间：2024-06-06 01:45:02

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeSubscribeJobs
  - 新增入参：Topic

第 28 次发布

发布时间：2024-04-10 01:14:34

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ConfigureSyncJob
  - 新增入参：SrcConnectType

修改数据结构：

- DBEndpointInfo
  - 新增成员：ConnectType

第 27 次发布

发布时间：2024-03-29 01:16:13

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Endpoint
  - 新增成员：ChildInstanceId, ChildInstanceType
- EndpointItem
  - 新增成员：ChildInstanceId, ChildInstanceType

第 26 次发布

发布时间：2024-03-27 01:15:59

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeSubscribeDetail
  - 新增出参：KafkaVersion

第 25 次发布

发布时间：2024-03-14 01:15:02

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DBInfo
  - 新增成员：SetId

第 24 次发布

发布时间：2024-03-08 01:15:05

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Options
  - 新增成员：FilterBeginCommit, FilterCheckpoint
- TopicRule
  - 新增成员：Columns

第 23 次发布

发布时间：2024-02-23 01:14:38

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- AdvancedObjectsItem
- DifferenceAdvancedObjectsDetail
- DifferenceData
- DifferenceDataDetail
- DifferenceRowDetail
- RowsCountDifference

修改数据结构：

- CompareDetailInfo
  - 新增成员：DifferenceAdvancedObjects, DifferenceData, DifferenceRow

第 22 次发布

发布时间：2024-01-12 01:14:12

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- MigrateOption
  - 新增成员：MigrateWay

第 21 次发布

发布时间：2024-01-10 01:13:50

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ConfigureSubscribeJob
- CreateConsumerGroup
- CreateSubscribe
- CreateSubscribeCheckJob
- DeleteConsumerGroup
- DescribeConsumerGroups
- DescribeOffsetByTime
- DescribeSubscribeCheckJob
- DescribeSubscribeDetail
- DescribeSubscribeJobs
- DescribeSubscribeReturnable
- DestroyIsolatedSubscribe
- IsolateSubscribe
- ModifyConsumerGroupDescription
- ModifyConsumerGroupPassword
- ModifySubscribeAutoRenewFlag
- ModifySubscribeName
- ModifySubscribeObjects
- ResetConsumerGroupOffset
- ResetSubscribe
- ResumeSubscribe
- StartSubscribe

新增数据结构：

- DistributeRule
- EndpointItem
- GroupInfo
- ModifiedSubscribeObject
- MonitorInfo
- OffsetTimeMap
- PartitionAssignment
- PipelineInfo
- SubsErr
- SubscribeCheckStepInfo
- SubscribeCheckStepTip
- SubscribeInfo
- SubscribeKafkaConfig
- SubscribeObject

第 20 次发布

发布时间：2023-12-27 01:17:28

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ErrInfo
  - **修改成员：** Reason, Message, Solution
- JobItem
  - **修改成员：** DumperResumeCtrl

第 19 次发布

发布时间：2023-11-15 01:36:06

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ResumeMigrateJob
  - **修改入参：** ResumeOption

第 18 次发布

发布时间：2023-10-23 15:36:37

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ModifyMigrateRuntimeAttribute

第 17 次发布

发布时间：2023-10-10 01:36:36

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SyncJobInfo
  - **修改成员：** SrcNodeType, SrcInfos, DstNodeType, DstInfos, DumperResumeCtrl

第 16 次发布

发布时间：2023-09-01 01:13:57

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- Column
- CompareColumnItem

修改数据结构：

- CompareTableItem
  - 新增成员：ColumnMode, Columns
- Table
  - 新增成员：ColumnMode, Columns

第 15 次发布

发布时间：2023-08-18 16:50:40

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ModifyMigrateRateLimit
- ModifySyncRateLimit

第 14 次发布

发布时间：2023-08-09 01:23:41

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeMigrationDetail
  - 新增出参：DumperResumeCtrl, RateLimitOption

新增数据结构：

- ErrInfo
- RateLimitOption

修改数据结构：

- Endpoint
  - 新增成员：CcnOwnerUin
- JobItem
  - 新增成员：DumperResumeCtrl
- Options
  - 新增成员：RateLimitOption, AutoRetryTimeRangeMinutes
- SyncDetailInfo
  - 新增成员：ErrInfo
  - **修改成员：** CauseOfCompareDisable
- SyncJobInfo
  - 新增成员：SrcNodeType, SrcInfos, DstNodeType, DstInfos, DumperResumeCtrl

第 13 次发布

发布时间：2023-05-29 14:42:50

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateModifyCheckSyncJob
- DescribeModifyCheckSyncJobResult
- ModifySyncJobConfig
- StartModifySyncJob

新增数据结构：

- DynamicOptions

第 12 次发布

发布时间：2023-03-14 01:22:11

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ConfigureSyncJob
  - 新增入参：SrcInfos, SrcNodeType, DstInfos, DstNodeType

新增数据结构：

- KafkaOption
- SyncDBEndpointInfos
- TopicRule

修改数据结构：

- Options
  - 新增成员：KafkaOption

第 11 次发布

发布时间：2023-02-27 01:22:55

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ConfigureSyncJob
  - **修改入参：** Options

第 10 次发布

发布时间：2023-02-22 01:24:05

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DBEndpointInfo
  - 新增成员：DatabaseNetEnv

第 9 次发布

发布时间：2023-02-16 01:24:00

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Endpoint
  - 新增成员：DatabaseNetEnv
- SyncDetailInfo
  - 新增成员：CauseOfCompareDisable

第 8 次发布

发布时间：2023-02-10 01:32:00

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- OnlineDDL
  - 新增成员：Status

- Table
  - 新增成员：TmpTables, TableEditMode

第 7 次发布

发布时间：2023-01-12 01:17:34

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- SkipCheckItem
  - 新增入参：ForeignKeyFlag
  - 新增出参：Message

第 6 次发布

发布时间：2023-01-05 01:19:52

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ContinueMigrateJob
- ContinueSyncJob
- PauseMigrateJob

修改接口：

- CreateCompareTask
  - 新增入参：Options
- DescribeCompareTasks
  - 新增入参：CompareTaskId, Status
- ModifyCompareTask
  - 新增入参：Options

新增数据结构：

- CompareOptions

修改数据结构：

- CompareAbstractInfo
  - 新增成员：Options, Objects, NearlyTableCount, SrcSampleRows, DstSampleRows, StartedAt, FinishedAt
- CompareObject
  - 新增成员：AdvancedObjects
- CompareTaskItem
  - 新增成员：Method, Options, Message

第 5 次发布

发布时间：2023-01-04 01:20:09

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- PauseSyncJob

第 4 次发布

发布时间：2022-12-23 01:28:44

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ConfigureSyncJob
  - 新增入参：JobMode, AutoRetryTimeRangeMinutes
  - **修改入参：** SrcInfo, DstInfo
- ModifyMigrationJob
  - 新增入参：AutoRetryTimeRangeMinutes

新增数据结构：

- OnlineDDL

修改数据结构：

- Database
  - 新增成员：TriggerMode, Triggers, EventMode, Events
- Endpoint
  - 新增成员：Role, EncryptConn
- JobItem
  - 新增成员：AutoRetryTimeRangeMinutes
- Objects
  - 新增成员：OnlineDDL
- SyncJobInfo
  - 新增成员：AutoRetryTimeRangeMinutes

第 3 次发布

发布时间：2022-12-12 01:23:41

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- SkipCheckItem
- SkipSyncCheckItem

第 2 次发布

发布时间：2022-12-05 06:45:09

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Endpoint
  - 新增成员：RoleExternalId
- StepTip
  - 新增成员：SkipInfo
- SyncJobInfo
  - 新增成员：TradeStatus, InstanceClass, AutoRenew, OfflineTime

第 1 次发布

发布时间：2022-10-28 16:26:38

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CompleteMigrateJob
- ConfigureSyncJob
- CreateCheckSyncJob
- CreateCompareTask
- CreateMigrateCheckJob
- CreateMigrationService
- CreateSyncJob
- DeleteCompareTask
- DescribeCheckSyncJobResult
- DescribeCompareReport
- DescribeCompareTasks
- DescribeMigrateDBInstances
- DescribeMigrationCheckJob
- DescribeMigrationDetail
- DescribeMigrationJobs
- DescribeSyncJobs
- DestroyMigrateJob
- DestroySyncJob
- IsolateMigrateJob
- IsolateSyncJob
- ModifyCompareTask
- ModifyCompareTaskName
- ModifyMigrateJobSpec
- ModifyMigrateName
- ModifyMigrationJob
- RecoverMigrateJob
- RecoverSyncJob
- ResizeSyncJob
- ResumeMigrateJob
- ResumeSyncJob
- StartCompare
- StartMigrateJob
- StartSyncJob
- StopCompare
- StopMigrateJob
- StopSyncJob

新增数据结构：

- CheckStep
- CheckStepInfo
- CompareAbstractInfo
- CompareDetailInfo
- CompareObject
- CompareObjectItem
- CompareTableItem
- CompareTaskInfo
- CompareTaskItem
- CompareViewItem
- ConflictHandleOption
- ConsistencyOption
- DBEndpointInfo
- DBInfo
- DBItem
- Database

- DatabaseTableObject
- DdlOption
- DetailCheckItem
- DifferenceDetail
- DifferenceItem
- Endpoint
- ErrorInfoItem
- JobItem
- KeyValuePairOption
- MigrateAction
- MigrateDBItem
- MigrateDetailInfo
- MigrateOption
- Objects
- Options
- ProcessProgress
- ProcessStepTip
- RoleItem
- SkippedDetail
- SkippedItem
- StepDetailInfo
- StepInfo
- StepTip
- SyncDetailInfo
- SyncJobInfo
- Table
- TableItem
- TagFilter
- TagItem
- TradeInfo
- View
- ViewItem

版权所有：腾讯云计算（北京）有限责任公司

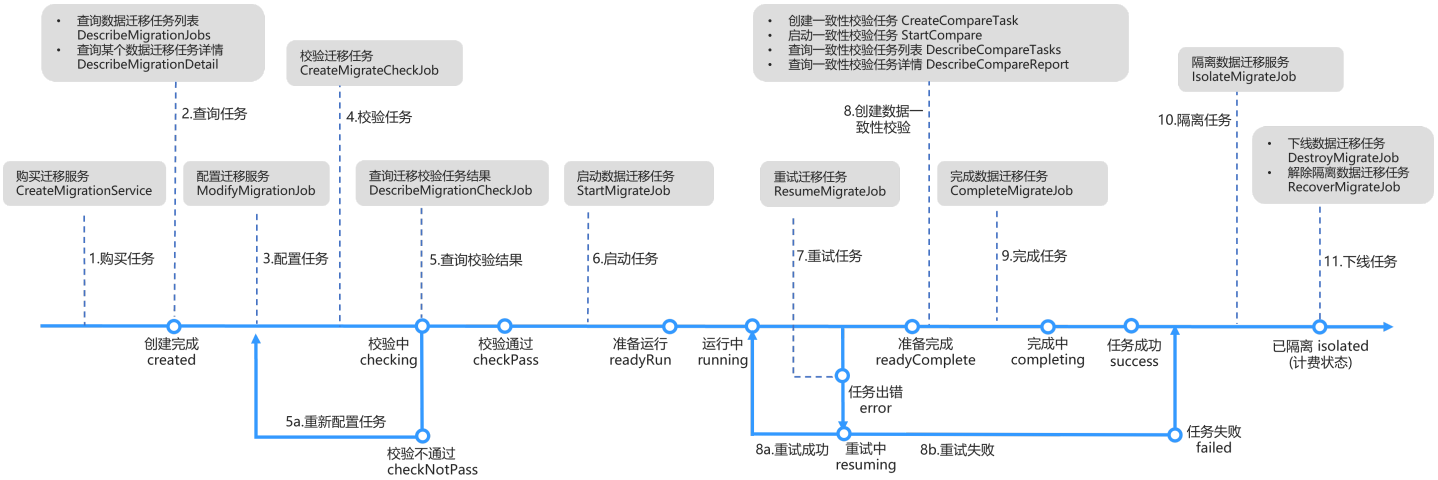
简介

最近更新时间：2025-04-25 01:29:25

数据传输服务（Data Transfer Service，DTS）提供数据迁移、数据同步、数据订阅于一体的数据库数据传输服务，帮助您在业务不停服的前提下轻松完成数据库迁移，利用实时同步通道轻松构建异地容灾的高可用数据库架构。

数据迁移 API 使用关键流程

数据迁移任务中使用的关键接口如下。圆圈表示迁移任务的不同状态，请按照图中序号，依次进行接口调用操作。



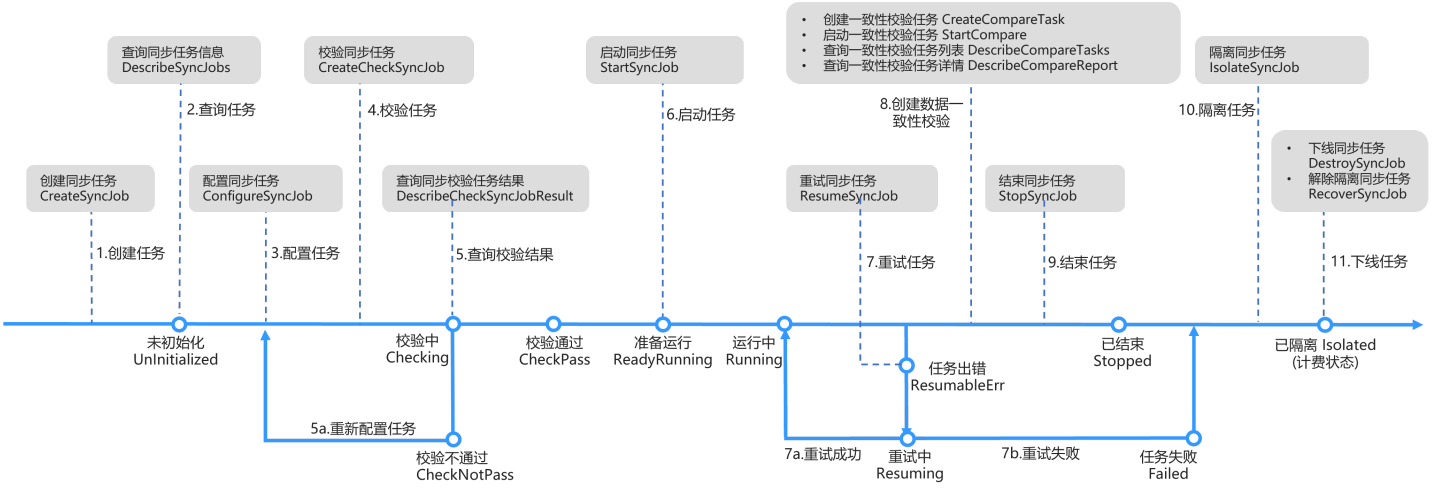
迁移任务中，各任务状态下支持的操作和相关接口如下表所示：

任务状态	支持的操作	关键接口	说明
–	–	购买迁移服务CreateMigrationService	购买迁移任务资源。购买完成后，会返回随机生成的迁移任务ID列表。
创建完成 created	配置	配置迁移服务 ModifyMigrationJob 校验迁移任务 CreateMigrateCheckJob 查询数据迁移任务列表 DescribeMigrationJobs	购买操作完成，下一步进行迁移任务配置。 任务配置完成后，发起校验。任务校验即在迁移任务启动前，对源库和目标库的各项参数进行校验，校验通过后才能启动迁移任务。 可通过“查询数据迁移任务列表 DescribeMigrationJobs”查询所有迁移任务列表，获取任务ID。
校验中 checking	–	查询迁移校验任务结果 DescribeMigrationCheckJob	迁移任务校验进行中。
校验通过 checkPass	立即启动、配置、校验	启动数据迁移任务 StartMigrateJob	迁移任务校验通过，启动数据迁移任务。
校验不通过 checkNotPass	配置、校验	配置迁移服务 ModifyMigrationJob	迁移任务校验不通过，需要重新配置迁移任务，然后再次发起校验。
准备运行 readyRun	终止	–	进行**启动**操作后，准备启动数据迁移任务的过渡状态。
运行中 running	终止、创建一致性校验任务（仅支持该功能的链路可操作）	终止数据迁移任务 StopMigrateJob	任务正在运行中。 用户如果不需要继续迁移，可以通过**终止**操作来取消任务。任务终止后不可恢复。
终止中 stopping	–	–	进行**终止**操作后的过渡状态。
已终止 canceled	–	隔离数据迁移服务 IsolateMigrateJob	已终止的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
准备完成 readyComplete	完成、终止、创建一致性校验任务（仅支持该功	完成数据迁移任务 CompleteMigrateJob 创建一致性校验任务 CreateCompareTask	选择“全量+增量迁移“场景，当任务状态为”准备完成“时，表明全量和增量数据都已迁移完成，需要用户选择一个合适的时间，通过**完成**操作来结束任务。 建议在**完成**操作前先进行数据一致性校验。 数据一致性校验，即对比源库中选择迁移的数据，和已迁

任务状态	支持的操作	关键接口	说明
	能的链路可操作）	启动一致性校验任务 StartCompare 查询一致性校验任务列表 DescribeCompareTasks 查询一致性校验任务详情 DescribeCompareReport	移到目标库中的数据结果是否一致，并查看不一致的数据详情。创建一致性校验任务成功后，会返回一致性校验的任务 ID。
完成中 completing	–	–	进行**完成**操作后的过渡状态。
任务成功 success	–	隔离数据迁移服务 IsolateMigrateJob	迁移任务顺利完成。已完成的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
任务出错 error	终止、重试	重试迁移任务 ResumeMigrateJob	迁移过程中，出现异常导致任务中断，任务状态为“error”时，可以进行任务重试。
重试中 resuming	–	–	进行**重试**操作后的过渡状态，重试成功则任务正常进行，重试失败则任务状态为“任务失败 failed”。
任务失败 failed	重试（仅 Redis 迁移可进行重试操作）	隔离数据迁移服务 IsolateMigrateJob	迁移任务失败。 失败后的任务需要进行隔离操作，以便及时解冻费用（按量计费会冻结一小时费用）。
以上所有状态	查看、变配、隔离	查询某个数据迁移任务详情 DescribeMigrationDetail 调整实例规格 ModifyMigrateJobSpec 隔离数据迁移服务 IsolateMigrateJob	通过**查看**操作查看某个指定迁移任务的详情。 通过**变配**操作调整迁移链路的规格，不同规格的传输速率不同。 通过**隔离**操作将已结束的任务、失败的任务、不需要的任务，进行隔离。
已隔离 isolated (计费状态)	解隔离（恢复）、下线	解除隔离数据迁移任务 RecoverMigrateJob 下线数据迁移任务 DestroyMigrateJob	用户主动发起隔离操作，或者任务在欠费1天后，都会进入隔离状态，隔离中的资源会保留7天，业务停服不可用，隔离7天后资源会被清除。 已隔离的任务，用户可以主动发起下线操作，也可以等隔离7天后系统自动下线。 如果用户误操作了隔离，可以通过**解隔离**操作，将任务恢复到隔离前的状态。

数据同步 API 使用关键流程

数据同步任务中使用的关键接口如下。圆圈表示同步任务的不同状态，请按照图中序号，依次进行接口调用操作。



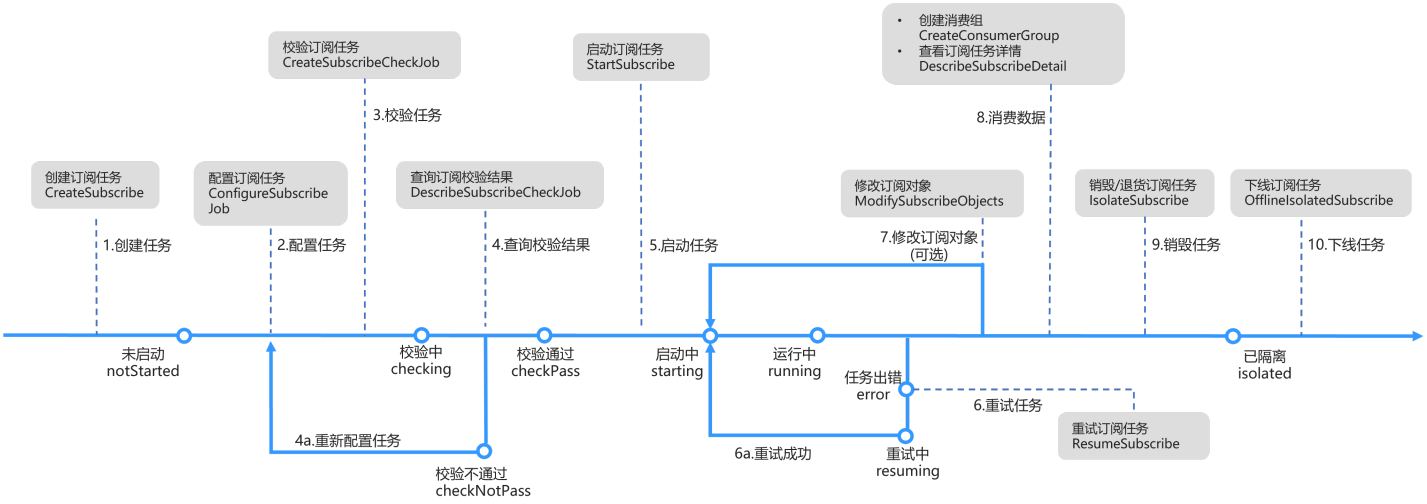
同步任务中，各任务状态下支持的操作和相关接口如下表所示：

任务状态	支持的操作	关键接口	说明
–	–	创建同步服务 CreateSyncJob	购买同步任务资源。购买完成后，会返回随机生成的同步任务ID列表。
未初始化 UnInitialized	配置	配置同步任务 ConfigureSyncJob 校验同步任务 CreateCheckSyncJob 查询同步任务信息 DescribeSyncJobs	购买操作完成，下一步进行同步任务配置。 任务配置完成后，发起校验。任务校验即在同步任务启动前，对源库和目标库的各项参数进行校验，校验通过后才能启动同步任务。 可通过“查询同步任务信息 DescribeSyncJobs”查询所有同步任务列表，获取任务ID。

任务状态	支持的操作	关键接口	说明
校验中 Checking	–	查询同步校验任务结果 DescribeCheckSyncJobResult	同步任务校验进行中。
校验通过 CheckPass	启动、配置、校验	启动同步任务 StartSyncJob	同步任务校验通过，启动数据同步任务。
校验不通过 CheckNotPass	配置、校验	配置同步任务 ConfigureSyncJob	同步任务校验不通过，需要重新配置同步任务，然后再次发起校验。
准备运行 ReadyRunning	结束	–	启动操作后，准备启动数据同步任务的过渡状态。
运行中 Running	结束	结束同步任务 StopSyncJob	任务正在运行中。 用户如果不需要继续同步，可以通过**结束**操作来停止任务。
结束中 Stopping	–	–	进行**结束**操作后的过渡状态。
已结束 Stopped	–	隔离同步任务 IsolateSyncJob	同步任务顺利完成。已完成的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
任务出错 ResumableErr	结束、重试	重试同步任务 ResumeSyncJob	同步过程中，出现异常导致任务中断，任务状态为“ResumableErr”时，可以进行任务重试。
重试中 Resuming	–	–	进行**重试**操作后的过渡状态，重试成功则任务正常进行，重试失败则任务状态为“任务失败 Failed”。
任务失败 Failed	–	隔离同步任务 IsolateSyncJob	同步任务失败。 失败后的任务需要进行隔离操作，以便及时解冻费用（按量计费会冻结一小时费用）。
以上所有状态	查看、变配、隔离	查询同步任务信息 DescribeSyncJobs 调整同步任务规格 ResizeSyncJob 隔离同步任务 IsolateSyncJob	通过**查看**操作查看某个指定同步任务的详情。 通过**变配**操作调整同步链路的规格，不同规格的传输速率不同。 通过**隔离**操作将已结束的任务、失败的任务、不需要的任务，进行隔离。
已隔离 Isolated (计费状态)	解隔离（恢复）、下线	解除隔离同步任务 RecoverSyncJob 下线同步任务 DestroySyncJob	用户主动发起隔离操作，或者任务在欠费1天后，都会进入隔离状态，隔离中的资源会保留7天，业务停服不可用，隔离7天后资源会被清除。 已隔离的任务，用户可以主动发起下线操作，也可以等隔离7天后系统自动下线。 如果用户误操作了隔离，可以通过**解隔离**操作，将任务恢复到隔离前的状态。

数据订阅 API 使用关键流程

数据订阅任务中使用的关键接口如下。圆圈表示订阅任务的不同状态，请按照图中序号，依次进行接口调用操作。



任务状态	支持的操作	关键接口	说明
–	–	创建订阅任务 CreateSubscribe	购买订阅任务资源。购买完成后，会返回随机生成的订阅任务ID列表。

任务状态	支持的操作	关键接口	说明
未启动 notStarted	配置订阅、销毁/退货	配置订阅任务 ConfigureSubscribe 校验订阅任务 CreateSubscribeCheckJob	购买完成，下一步进行订阅任务配置。 任务配置完成后，发起校验。校验任务即在订阅任务启动前，对源库的各项参数进行校验，校验通过后才能启动订阅任务。
校验中 checking		查询订阅校验结果 DescribeSubscribeCheckJob	订阅任务校验进行中。
校验通过 checkPass	启动、配置订阅、校验、销毁/退货	启动校验任务 StartSubscribe	校验通过，启动数据订阅任务。
校验不通过 checkNotPass	配置订阅、校验、销毁/退货	配置订阅任务 ConfigureSubscribeJob	订阅任务校验不通过，需要重新配置订阅任务，然后再次发起校验。
启动中 starting	销毁/退货	-	订阅任务启动中，过渡状态。
任务运行 running	修改订阅对象、重置订阅、销毁/退货、查看任务详情	创建消费组 CreateConsumerGroup 修改订阅对象 ModifySubscribeObjects 重置订阅任务 ResetSubscribe 销毁/退货订阅任务 IsolateSubscribe 查看订阅任务详情 DescribeSubscribeDetail	订阅任务运行中，此时可以创建消费组消费Kafka。 修改订阅对象（ModifySubscribeObjects），是对正在进行的任务，追加或者删除对象，Kafka 分区规则等。 销毁任务，是停止任务并清空配置和kafka数据，同时进行计费清算。 重置订阅任务，是对不再需要的任务进行重置，旧的任务会停止并清空历史配置和kafka数据，然后重新配置新任务。重置任务可以避免用户先退费旧任务，再购买新任务，频繁地进行购买操作。 任务运行中可查看任务详情并操作kafka相关接口。
停止中 stopping	-		重置订阅对象时，订阅任务停止中。重置后待任务状态变为“未启动 notStarted”状态才可重新配置任务。
任务出错 error	恢复、销毁/退货	恢复订阅任务 ResumeSubscribe	订阅过程中，如果任务出现异常导致任务中断，状态为“error”时，可以尝试恢复任务。
已隔离 isolated	下线	下线已隔离的订阅任务 OfflineIsolatedSubscribe	销毁/退货操作后，任务会进入隔离的状态。隔离的任务，用户可以主动发起下线操作，也可以等隔离1天后系统自动下线。

API 概览

最近更新时间：2024-08-02 01:32:53

（新版）数据迁移相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeMigrationJobs	查询数据迁移任务列表	20
CreateMigrationService	购买迁移任务	20
ModifyMigrationJob	配置迁移服务	20
CreateMigrateCheckJob	校验迁移任务	20
DescribeMigrationCheckJob	查询迁移校验任务结果	20
SkipCheckItem	跳过迁移校验检查项	20
StartMigrateJob	启动数据迁移任务	20
CompleteMigrateJob	完成数据迁移任务	20
ContinueMigrateJob	恢复暂停中的迁移任务	20
DescribeMigrateDBInstances	查询可迁移的实例列表	20
DescribeMigrationDetail	查询某个迁移任务详情	20
DestroyMigrateJob	下线数据迁移任务	20
IsolateMigrateJob	隔离数据迁移任务	20
ModifyMigrateJobSpec	调整实例规格	20
ModifyMigrateName	修改迁移名称	20
ModifyMigrateRuntimeAttribute	修改任务运行时属性	20
PauseMigrateJob	暂停迁移任务	20
RecoverMigrateJob	解除隔离数据迁移任务	20
ResumeMigrateJob	重试迁移任务	20
StopMigrateJob	终止数据迁移任务	20
ModifyMigrateRateLimit	修改迁移任务的传输速率	20

数据同步相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeSyncJobs	查询同步任务信息	100
CreateSyncJob	创建同步任务	20
ConfigureSyncJob	配置同步任务	20
CreateCheckSyncJob	校验同步任务	20
DescribeCheckSyncJobResult	查询同步校验任务结果	20
SkipSyncCheckItem	跳过同步校验检查项	20
StartSyncJob	启动同步任务	20
IsolateSyncJob	隔离同步任务	20
RecoverSyncJob	解除隔离同步任务	20

版权所有：腾讯云计算（北京）有限责任公司

接口名称	接口功能	频率限制（次/秒）
PauseSyncJob	暂停同步任务	20
ContinueSyncJob	恢复被暂停的同步任务	20
ResumeSyncJob	重试同步任务	20
StopSyncJob	结束同步任务	20
DestroySyncJob	下线同步任务	20
ResizeSyncJob	调整同步任务规格	20
ModifySyncJobConfig	修改同步任务配置	20
CreateModifyCheckSyncJob	创建修改同步配置的校验任务	20
DescribeModifyCheckSyncJobResult	查询修改对象的校验任务的结果	20
StartModifySyncJob	开始修改配置流程	20
ModifySyncRateLimit	修改同步任务的传输速率	20

数据一致性校验相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeCompareReport	查询一致性校验任务详情	20
StopCompare	终止一致性校验任务	20
DeleteCompareTask	删除一致性校验任务	20
ModifyCompareTask	修改一致性校验任务	20
ModifyCompareTaskName	修改一致性校验任务名称	20
CreateCompareTask	创建一致性校验任务	20
StartCompare	启动一致性校验任务	20
DescribeCompareTasks	查询一致性校验任务列表	20

（新版）数据订阅相关接口

接口名称	接口功能	频率限制（次/秒）
CreateSubscribe	创建订阅任务	20
ConfigureSubscribeJob	配置订阅任务	20
CreateSubscribeCheckJob	校验订阅任务	20
DescribeSubscribeCheckJob	查询订阅校验结果	20
StartSubscribe	启动订阅任务	20
ModifySubscribeObjects	修改订阅对象	20
DescribeSubscribeDetail	查询订阅任务详情	20
DescribeSubscribeJobs	获取数据订阅实例列表	100
ResumeSubscribe	重试订阅任务	20
CreateConsumerGroup	创建消费组	20
DescribeConsumerGroups	获取数据订阅的消费组详情	20
ModifyConsumerGroupDescription	修改消费组备注	20

版权所有：腾讯云计算（北京）有限责任公司

接口名称	接口功能	频率限制（次/秒）
ModifyConsumerGroupPassword	修改消费组密码	20
DescribeOffsetByTime	查询kafka offset	20
ResetConsumerGroupOffset	重置kafka offset	20
DeleteConsumerGroup	删除消费组	20
ResetSubscribe	重置订阅任务	20
ModifySubscribeAutoRenewFlag	修改订阅实例自动续费标识	20
ModifySubscribeName	修改数据订阅实例的名称	20
DescribeSubscribeReturnable	查询订阅实例是否可以退换	20
IsolateSubscribe	销毁订阅任务	20
DestroyIsolatedSubscribe	下线已隔离的数据订阅任务	20

注意：

以上给出的接口频率限制维度为 `API` + **接入地域** + **子账号，有关限频更多说明参考：** API 频率限制说明

调用方式
请求结构

最近更新时间：2025-05-21 01:17:53

1. 服务地址

API 支持就近地域接入，本产品就近地域接入域名为 dts.tencentcloudapi.com ，也支持指定地域域名访问，例如广州地域的域名为 dts.ap-guangzhou.tencentcloudapi.com 。

推荐使用就近地域接入域名。根据调用接口时客户端所在位置，会自动解析到**最近的**某个具体地域的服务器。例如在广州发起请求，会自动解析到广州的服务器，效果和指定 dts.ap-guangzhou.tencentcloudapi.com 是一致的。

注意：对时延敏感的业务，建议指定带地域的域名。

注意：域名是 API 的接入点，并不代表产品或者接口实际提供服务的地域。产品支持的地域列表请在调用方式/公共参数文档中查阅，接口支持的地域请在接口文档输入参数中查阅。

目前支持的域名列表为：

接入地域	域名
就近地域接入（推荐，只支持非金融区）	dts.tencentcloudapi.com
华南地区（广州）	dts.ap-guangzhou.tencentcloudapi.com
华东地区（上海）	dts.ap-shanghai.tencentcloudapi.com
华东地区（南京）	dts.ap-nanjing.tencentcloudapi.com
华北地区（北京）	dts.ap-beijing.tencentcloudapi.com
西南地区（成都）	dts.ap-chengdu.tencentcloudapi.com
西南地区（重庆）	dts.ap-chongqing.tencentcloudapi.com
港澳台地区（中国香港）	dts.ap-hongkong.tencentcloudapi.com
亚太东南（新加坡）	dts.ap-singapore.tencentcloudapi.com
亚太东南（雅加达）	dts.ap-jakarta.tencentcloudapi.com
亚太东南（曼谷）	dts.ap-bangkok.tencentcloudapi.com
亚太东北（首尔）	dts.ap-seoul.tencentcloudapi.com
亚太东北（东京）	dts.ap-tokyo.tencentcloudapi.com
美国东部（弗吉尼亚）	dts.na-ashburn.tencentcloudapi.com
美国西部（硅谷）	dts.na-siliconvalley.tencentcloudapi.com
南美地区（圣保罗）	dts.sa-saopaulo.tencentcloudapi.com
欧洲地区（法兰克福）	dts.eu-frankfurt.tencentcloudapi.com

注意：由于金融区和非金融区是隔离不互通的，因此当访问金融区服务时（公共参数 Region 为金融区地域），需要同时指定带金融区地域的域名，最好和 Region 的地域保持一致。

金融区接入地域	金融区域名
华东地区（上海金融）	dts.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	dts.ap-shenzhen-fsi.tencentcloudapi.com

2. 通信协议

腾讯云 API 的所有接口均通过 HTTPS 进行通信，提供高安全性的通信通道。

3. 请求方法

支持的 HTTP 请求方法:

- POST（推荐）
- GET

POST 请求支持的 Content-Type 类型:

- application/json（推荐），必须使用签名方法 v3（TC3-HMAC-SHA256）。
- application/x-www-form-urlencoded，必须使用签名方法 v1（HmacSHA1 或 HmacSHA256）。
- multipart/form-data（仅部分接口支持），必须使用签名方法 v3（TC3-HMAC-SHA256）。

GET 请求的请求包大小不得超过32KB。POST 请求使用签名方法 v1（HmacSHA1、HmacSHA256）时不得超过1MB。POST 请求使用签名方法 v3（TC3-HMAC-SHA256）时支持10MB。

4. 字符编码

均使用 `UTF-8` 编码。

公共参数

最近更新时间：2025-03-28 01:31:44

公共参数是用于标识用户和接口签名的参数，如非必要，在每个接口单独的文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

公共参数的具体内容会因您使用的签名方法版本不同而有所差异。

使用签名方法 v3 的公共参数

签名方法 v3（有时也称作 TC3-HMAC-SHA256）相比签名方法 v1（有些文档可能会简称签名方法），更安全，支持更大的请求包，支持 POST JSON 格式，性能有一定提升，推荐使用该签名方法计算签名。完整介绍详见 签名方法 v3。

注意：出于简化的目的，部分接口文档中的示例使用的是签名方法 v1 GET 请求，而不是更安全的签名方法 v3。

使用签名方法 v3 时，公共参数需要统一放到 HTTP Header 请求头部中，如下表所示：

参数名称	类型	必选	描述
Action	String	是	HTTP 请求头：X-TC-Action。操作的接口名称。取值参考接口文档输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	-	HTTP 请求头：X-TC-Region。地域参数，用来标识希望操作哪个地域的数据。取值参考接口文档中输入参数章节关于公共参数 Region 的说明。**注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。**
Timestamp	Integer	是	HTTP 请求头：X-TC-Timestamp。当前 UNIX 时间戳，可记录发起 API 请求的时间。例如 1529223702。**注意：如果与服务器时间相差超过5分钟，会引起签名过期错误。**
Version	String	是	HTTP 请求头：X-TC-Version。操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
Authorization	String	是	HTTP 标准身份认证头部字段，例如： TC3-HMAC-SHA256 Credential=AKID***/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 其中， - TC3-HMAC-SHA256：签名方法，目前固定取该值； - Credential：签名凭证，AKID*** 是 SecretId；Date 是 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为具体产品名，通常为域名前缀。例如，域名 cvm.tencentcloudapi.com 意味着产品名是 cvm。本产品取值为 dts；tc3_request 为固定字符串； - SignedHeaders：参与签名计算的头部信息，content-type 和 host 为必选头部； - Signature：签名摘要，计算过程详见 文档。
Token	String	否	HTTP 请求头：X-TC-Token。即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	HTTP 请求头：X-TC-Language。指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表中的前十个，接口参数设置为偏移量 Offset=0，返回数量 Limit=10，则其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: ap-guangzhou
```

HTTP POST（application/json）请求结构示例：

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: application/json
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

{"Offset":0,"Limit":10}
```

HTTP POST（multipart/form-data）请求结构示例（仅特定的接口支持）：

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: multipart/form-data; boundary=58731222010402
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

--58731222010402
Content-Disposition: form-data; name="Offset"

0
--58731222010402
Content-Disposition: form-data; name="Limit"

10
--58731222010402--
```

使用签名方法 v1 的公共参数

使用签名方法 v1（有时会称作 HmacSHA256 和 HmacSHA1），公共参数需要统一放到请求串中，完整介绍详见文档

参数名称	类型	必选	描述
Action	String	是	操作的接口名称。取值参考接口文档中输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	-	地域参数，用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。**注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。**
Timestamp	Integer	是	当前 UNIX 时间戳，可记录发起 API 请求的时间。例如1529223702，如果与当前时间相差过大，会引起签名过期错误。
Nonce	Integer	是	随机正整数，与 Timestamp 联合起来，用于防止重放攻击。
SecretId	String	是	在 云API密钥 上申请的标识身份的 SecretId，一个 SecretId 对应唯一的 SecretKey ，而 SecretKey 会用来生成请求签名 Signature。
Signature	String	是	请求签名，用来验证此次请求的合法性，需要用户根据实际的输入参数计算得出。具体计算方法参见 文档。

参数名称	类型	必选	描述
Version	String	是	操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
SignatureMethod	String	否	签名方式，目前支持 HmacSHA256 和 HmacSHA1。只有指定此参数为 HmacSHA256 时，才使用 HmacSHA256 算法验证签名，其他情况均使用 HmacSHA1 验证签名。
Token	String	否	即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表，其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID********************************

Host: cvm.tencentcloudapi.com
```

HTTP POST 请求结构示例：

```
https://cvm.tencentcloudapi.com/

Host: cvm.tencentcloudapi.com
Content-Type: application/x-www-form-urlencoded

Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****************************
```

地域列表

本产品所有接口 Region 字段的可选值如下表所示。如果接口不支持该表中的所有地域，则会在接口文档中单独说明。

地域	取值
亚太东南（曼谷）	ap-bangkok
华北地区（北京）	ap-beijing
西南地区（成都）	ap-chengdu
西南地区（重庆）	ap-chongqing
华南地区（广州）	ap-guangzhou
港澳台地区（中国香港）	ap-hongkong
亚太东南（雅加达）	ap-jakarta
华东地区（南京）	ap-nanjing
亚太东北（首尔）	ap-seoul
华东地区（上海）	ap-shanghai
华东地区（上海金融）	ap-shanghai-fsi
华南地区（深圳金融）	ap-shenzhen-fsi
亚太东南（新加坡）	ap-singapore

地域	取值
亚太东北（东京）	ap-tokyo
欧洲地区（法兰克福）	eu-frankfurt
美国东部（弗吉尼亚）	na-ashburn
美国西部（硅谷）	na-siliconvalley
南美地区（圣保罗）	sa-saopaulo

签名方法 v3

最近更新时间：2024-12-25 01:31:55

以下文档说明了签名方法 v3 的签名过程，但仅在您编写自己的代码来调用腾讯云 API 时才有用。我们推荐您使用 腾讯云 API Explorer ，腾讯云 SDK 和 腾讯云命令行工具（TCCLI）等开发者工具，从而无需学习如何对 API 请求进行签名。

推荐使用 API Explorer　　点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个请求进行身份验证，用户需要使用安全凭证，经过特定的步骤对请求进行签名（Signature），每个请求都需要在公共参数中指定该签名结果并以指定的方式和格式发送请求。

为什么要进行签名

签名通过以下方式帮助保护请求：

1. 验证请求者的身份
   签名确保请求是由持有有效访问密钥的人发送的。请参阅控制台 云 API 密钥 页面获取密钥相关信息。

2. 保护传输中的数据
   为了防止请求在传输过程中被篡改，腾讯云 API 会使用请求参数来计算请求的哈希值，并将生成的哈希值加密后作为请求的一部分，发送到腾讯云 API 服务器。服务器会使用收到的请求参数以同样的过程计算哈希值，并验证请求中的哈希值。如果请求被篡改，将导致哈希值不一致，腾讯云 API 将拒绝本次请求。

签名方法 v3（TC3-HMAC-SHA256）功能上覆盖了以前的签名方法 v1，而且更安全，支持更大的请求，支持 JSON 格式，POST 请求支持传空数组和空字符串，性能有一定提升，推荐使用该签名方法计算签名。

首次接触，建议使用 API Explorer 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v3”，可以对生成签名过程进行验证，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 Python、Java、PHP、Go、NodeJS、.NET、C++、Ruby。

申请安全凭证

本文使用的安全凭证为密钥，密钥包括 SecretId 和 SecretKey。每个用户最多可以拥有两对密钥。

- SecretId：用于标识 API 调用者身份，可以简单类比为用户名。
- SecretKey：用于验证 API 调用者的身份，可以简单类比为密码。
- **用户必须严格保管安全凭证，避免泄露，否则将危及财产安全。如已泄露，请立刻禁用该安全凭证。**

申请安全凭证的具体步骤如下：

1. 登录 腾讯云管理中心控制台 。
2. 前往 云API密钥 的控制台页面。
3. 在 云API密钥 页面，单击【新建密钥】创建一对密钥。

签名版本 v3 签名过程

云 API 支持 GET 和 POST 请求。对于GET方法，只支持 `Content-Type: application/x-www-form-urlencoded` 协议格式。对于POST方法，目前支持 `Content-Type: application/json` 以及 `Content-Type: multipart/form-data` 两种协议格式，json 格式绝大多数接口均支持，multipart 格式只有特定接口支持，此时该接口不能使用 json 格式调用，参考具体业务接口文档说明。推荐使用 POST 请求，因为两者的结果并无差异，但 GET 请求只支持 32 KB 以内的请求包。

下面以云服务器查询广州区实例列表作为例子，分步骤介绍签名的计算过程。我们选择该接口是因为：

1. 云服务器默认已开通，该接口很常用；
2. 该接口是只读的，不会改变现有资源的状态；
3. 接口覆盖的参数种类较全，可以演示包含数据结构的数组如何使用。

在示例中，不论公共参数或者接口的参数，我们尽量选择容易犯错的情况。在实际调用接口时，请根据实际情况来，每个接口的参数并不相同，不要照抄这个例子的参数和值。此外，这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数（在 HTTP 头部设置，添加 X-TC- 前缀）。

假设用户的 SecretId 和 SecretKey 分别是：`AKID********************************` 和 `********************************`。用户想查看广州区云服务器名为“未命名”的主机状态，只返回一条数据。则请求可能为：

```
curl -X POST https://cvm.tencentcloudapi.com \
-H "Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f" \
-H "Content-Type: application/json; charset=utf-8" \
-H "Host: cvm.tencentcloudapi.com" \
-H "X-TC-Action: DescribeInstances" \
-H "X-TC-Timestamp: 1551113065" \
-H "X-TC-Version: 2017-03-12" \
-H "X-TC-Region: ap-guangzhou" \
-d '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
```

下面详细解释签名计算过程。

1. 拼接规范请求串

按如下伪代码格式拼接规范请求串（CanonicalRequest）：

```
CanonicalRequest =
HTTPRequestMethod + '\n' +
CanonicalURI + '\n' +
CanonicalQueryString + '\n' +
CanonicalHeaders + '\n' +
SignedHeaders + '\n' +
HashedRequestPayload
```

字段名称	解释
HTTPRequestMethod	HTTP 请求方法（GET、POST ）。此示例取值为 `POST`。
CanonicalURI	URI 参数，API 3.0 固定为正斜杠（/）。
CanonicalQueryString	发起 HTTP 请求 URL 中的查询字符串，对于 POST 请求，固定为空字符串""，对于 GET 请求，则为 URL 中问号（?）后面的字符串内容，例如：Limit=10&Offset=0。 注意：CanonicalQueryString 需要参考 RFC3986 进行 URLEncode 编码（特殊字符编码后需大写字母），字符集 UTF-8。推荐使用编程语言标准库进行编码。
CanonicalHeaders	参与签名的头部信息，至少包含 host 和 content-type 两个头部，也可加入其他头部参与签名以提高自身请求的唯一性和安全性，此示例额外增加了接口名头部。 拼接规则： 1. 头部 key 和 value 统一转成小写，并去掉首尾空格，按照 key:value\n 格式拼接； 2. 多个头部，按照头部 key（小写）的 ASCII 升序进行拼接。 此示例计算结果是 `content-type:application/json; charset=utf-8\nhost:cvm.tencentcloudapi.com\nx-tc-action:describeinstances\n`。 注意：content-type 必须和实际发送的相符合，有些编程语言网络库即使未指定也会自动添加 charset 值，如果签名时和发送时不一致，服务器会返回签名校验失败。
SignedHeaders	参与签名的头部信息，说明此次请求有哪些头部参与了签名，和 CanonicalHeaders 包含的头部内容是一一对应的。content-type 和 host 为必选头部。 拼接规则： 1. 头部 key 统一转成小写； 2. 多个头部 key（小写）按照 ASCII 升序进行拼接，并且以分号（;）分隔。 此示例为 content-type;host;x-tc-action
HashedRequestPayload	请求正文（payload，即 body，此示例为 `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`）的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(RequestPayload)))，即对 HTTP 请求正文做 SHA256 哈希，然后十六进制编

字段名称	解释
	码，最后编码串转换成小写字母。对于 GET 请求，RequestPayload 固定为空字符串。此示例计算结果是 `35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064`。

根据以上规则，示例中得到的规范请求串如下：

```
POST
/

content-type:application/json; charset=utf-8
host:cvm.tencentcloudapi.com
x-tc-action:describeinstances

content-type;host;x-tc-action
35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064
```

2. 拼接待签名字符串

按如下格式拼接待签名字符串：

```
StringToSign =
Algorithm + "\n" +
RequestTimestamp + "\n" +
CredentialScope + "\n" +
HashedCanonicalRequest
```

字段名称	解释
Algorithm	签名算法，目前固定为 `TC3-HMAC-SHA256`。
RequestTimestamp	请求时间戳，即请求头部的公共参数 X-TC-Timestamp 取值，取当前时间 UNIX 时间戳，精确到秒。此示例取值为 `1551113065`。
CredentialScope	凭证范围，格式为 Date/service/tc3_request，包含日期、所请求的服务和终止字符串（tc3_request）。**Date 为 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致**；service 为产品名，必须与调用的产品域名一致。此示例计算结果是 2019-02-25/cvm/tc3_request。
HashedCanonicalRequest	前述步骤拼接所得规范请求串的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(CanonicalRequest)))。此示例计算结果是 `7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84`。

注意：

1. Date 必须从时间戳 X-TC-Timestamp 计算得到，且时区为 UTC+0。如果加入系统本地时区信息，例如东八区，将导致白天和晚上调用成功，但是凌晨时调用必定失败。假设时间戳为 1551113065，在东八区的时间是 2019-02-26 00:44:25，但是计算得到的 Date 取 UTC+0 的日期应为 2019-02-25，而不是 2019-02-26。
2. Timestamp 必须是当前系统时间，且需确保系统时间和标准时间是同步的，如果相差超过五分钟则必定失败。如果长时间不和标准时间同步，可能运行一段时间后，请求失败，返回签名过期错误。

根据以上规则，示例中得到的待签名字符串如下：

```
TC3-HMAC-SHA256
1551113065
2019-02-25/cvm/tc3_request
7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84
```

3. 计算签名

1）计算派生签名密钥，伪代码如下：

```
SecretKey = "********************************"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

派生出的密钥 `SecretDate`、`SecretService` 和 `SecretSigning` 是二进制的数据，可能包含不可打印字符，将其转为十六进制字符串打印的输出分别为：
da98fb70dcf6b112dc21038d1eeeb3a95c74b4dcb12c1131f864f6066bd02be0，
8d70cbefb03939f929db64d32dc2ba89b1095620119fe3e050e2b18c5bd2752f，
b596b923aad85185e2d1f6659d2a062e0a86731226e021e61bfe06f7ed05f5af。

请注意，不同的编程语言，HMAC 库函数中参数顺序可能不一样，请以实际情况为准。此处的伪代码密钥参数 key 在前，消息参数 data 在后。通常标准库函数会提供二进制格式的返回值，也可能会提供打印友好的十六进制格式的返回值，此处使用的是二进制格式。

字段名称	解释
SecretKey	原始的 SecretKey，即 `********************************`。
Date	即 Credential 中的 Date 字段信息。此示例取值为 `2019-02-25`。
Service	即 Credential 中的 Service 字段信息。此示例取值为 `cvm`。

2）计算签名，伪代码如下：

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

此示例计算结果是 `10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f`。

4. 拼接 Authorization

按如下格式拼接 Authorization：

```
Authorization =
Algorithm + ' ' +
'Credential=' + SecretId + '/' + CredentialScope + ', ' +
'SignedHeaders=' + SignedHeaders + ', ' +
'Signature=' + Signature
```

字段名称	解释
Algorithm	签名方法，固定为 `TC3-HMAC-SHA256`。
SecretId	密钥对中的 SecretId，即 `AKID********************************`。
CredentialScope	见上文，凭证范围。此示例计算结果是 `2019-02-25/cvm/tc3_request`。
SignedHeaders	见上文，参与签名的头部信息。此示例取值为 `content-type;host;x-tc-action`。
Signature	签名值。此示例计算结果是 `10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f`。

根据以上规则，示例中得到的值为：

```
TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;hos
t;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
```

最终完整的调用信息如下：

```
POST https://cvm.tencentcloudapi.com/
Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=c
ontent-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
Content-Type: application/json; charset=utf-8
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1551113065
X-TC-Region: ap-guangzhou

{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}
```

注意：

请求发送时的 HTTP 头部（Header）和请求体（Payload）必须和签名计算过程中的内容完全一致，否则会返回签名不一致错误。可以通过打印实际请求内容，网络抓包等方式对比排查。

签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 SDK 中心。当前支持的编程语言有：

- Python
- Java
- PHP
- Go
- NodeJS
- .NET
- C++
- Ruby

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- Signature Demo

为了更清楚地解释签名过程，下面以实际编程语言为例，将上述的签名过程完整实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

Java

```
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPITC3Demo {
private final static Charset UTF8 = StandardCharsets.UTF_8;
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
private final static String SECRET_ID = System.getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
```

```
private final static String SECRET_KEY = System.getenv("TENCENTCLOUD_SECRET_KEY");
private final static String CT_JSON = "application/json; charset=utf-8";

public static byte[] hmac256(byte[] key, String msg) throws Exception {
Mac mac = Mac.getInstance("HmacSHA256");
SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
mac.init(secretKeySpec);
return mac.doFinal(msg.getBytes(UTF8));
}

public static String sha256Hex(String s) throws Exception {
MessageDigest md = MessageDigest.getInstance("SHA-256");
byte[] d = md.digest(s.getBytes(UTF8));
return DatatypeConverter.printHexBinary(d).toLowerCase();
}

public static void main(String[] args) throws Exception {
String service = "cvm";
String host = "cvm.tencentcloudapi.com";
String region = "ap-guangzhou";
String action = "DescribeInstances";
String version = "2017-03-12";
String algorithm = "TC3-HMAC-SHA256";
String timestamp = "1551113065";
//String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
// 注意时区，否则容易出错
sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

// ************* 步骤 1: 拼接规范请求串 *************
String httpRequestMethod = "POST";
String canonicalUri = "/";
String canonicalQueryString = "";
String canonicalHeaders = "content-type:application/json; charset=utf-8\n"
+ "host:" + host + "\n" + "x-tc-action:" + action.toLowerCase() + "\n";
String signedHeaders = "content-type;host;x-tc-action";

String payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}";
String hashedRequestPayload = sha256Hex(payload);
String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryString + "\n"
+ canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
System.out.println(canonicalRequest);

// ************* 步骤 2: 拼接待签名字符串 *************
String credentialScope = date + "/" + service + "/" + "tc3_request";
String hashedCanonicalRequest = sha256Hex(canonicalRequest);
String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
System.out.println(stringToSign);

// ************* 步骤 3: 计算签名 *************
byte[] secretDate = hmac256(("TC3" + SECRET_KEY).getBytes(UTF8), date);
byte[] secretService = hmac256(secretDate, service);
byte[] secretSigning = hmac256(secretService, "tc3_request");
String signature = DatatypeConverter.printHexBinary(hmac256(secretSigning, stringToSign)).toLowerCase();
System.out.println(signature);
```

```
// ************* 步骤 4: 拼接 Authorization *************
String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
System.out.println(authorization);

TreeMap<String, String> headers = new TreeMap<String, String>();
headers.put("Authorization", authorization);
headers.put("Content-Type", CT_JSON);
headers.put("Host", host);
headers.put("X-TC-Action", action);
headers.put("X-TC-Timestamp", timestamp);
headers.put("X-TC-Version", version);
headers.put("X-TC-Region", region);

StringBuilder sb = new StringBuilder();
sb.append("curl -X POST https://").append(host)
.append(" -H \"Authorization: ").append(authorization).append("\"")
.append(" -H \"Content-Type: application/json; charset=utf-8\"")
.append(" -H \"Host: ").append(host).append("\"")
.append(" -H \"X-TC-Action: ").append(action).append("\"")
.append(" -H \"X-TC-Timestamp: ").append(timestamp).append("\"")
.append(" -H \"X-TC-Version: ").append(version).append("\"")
.append(" -H \"X-TC-Region: ").append(region).append("\"")
.append(" -d '").append(payload).append("'");
System.out.println(sb.toString());
}
}
```

Python

```
# -*- coding: utf-8 -*-
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

service = "cvm"
host = "cvm.tencentcloudapi.com"
endpoint = "https://" + host
region = "ap-guangzhou"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
#timestamp = int(time.time())
timestamp = 1551113065
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 1, "Filters": [{"Values": [u"未命名"], "Name": "instance-name"}]}

# ************* 步骤 1: 拼接规范请求串 *************
http_request_method = "POST"
canonical_uri = "/"
canonical_querystring = ""
```

```
ct = "application/json; charset=utf-8"
payload = json.dumps(params)
canonical_headers = "content-type:%s\nhost:%s\nx-tc-action:%s\n" % (ct, host, action.lower())
signed_headers = "content-type;host;x-tc-action"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
canonical_uri + "\n" +
canonical_querystring + "\n" +
canonical_headers + "\n" +
signed_headers + "\n" +
hashed_request_payload)
print(canonical_request)

# ************* 步骤 2: 拼接待签名字符串 *************
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
str(timestamp) + "\n" +
credential_scope + "\n" +
hashed_canonical_request)
print(string_to_sign)


# ************* 步骤 3: 计算签名 *************
# 计算签名摘要函数
def sign(key, msg):
return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ************* 步骤 4: 拼接 Authorization *************
authorization = (algorithm + " " +
"Credential=" + secret_id + "/" + credential_scope + ", " +
"SignedHeaders=" + signed_headers + ", " +
"Signature=" + signature)
print(authorization)

print('curl -X POST ' + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + host + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + str(timestamp) + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + "'")
```

Golang

```
package main

import (
"crypto/hmac"
```

```
"crypto/sha256"
"encoding/hex"
"fmt"
"os"
"strings"
"time"
)

func sha256hex(s string) string {
b := sha256.Sum256([]byte(s))
return hex.EncodeToString(b[:])
}

func hmacsha256(s, key string) string {
hashed := hmac.New(sha256.New, []byte(key))
hashed.Write([]byte(s))
return string(hashed.Sum(nil))
}

func main() {
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
host := "cvm.tencentcloudapi.com"
algorithm := "TC3-HMAC-SHA256"
service := "cvm"
version := "2017-03-12"
action := "DescribeInstances"
region := "ap-guangzhou"
//var timestamp int64 = time.Now().Unix()
var timestamp int64 = 1551113065

// step 1: build canonical request string
httpRequestMethod := "POST"
canonicalURI := "/"
canonicalQueryString := ""
canonicalHeaders := fmt.Sprintf("content-type:%s\nhost:%s\nx-tc-action:%s\n",
"application/json; charset=utf-8", host, strings.ToLower(action))
signedHeaders := "content-type;host;x-tc-action"
payload := `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`
hashedRequestPayload := sha256hex(payload)
canonicalRequest := fmt.Sprintf("%s\n%s\n%s\n%s\n%s\n%s",
httpRequestMethod,
canonicalURI,
canonicalQueryString,
canonicalHeaders,
signedHeaders,
hashedRequestPayload)
fmt.Println(canonicalRequest)

// step 2: build string to sign
date := time.Unix(timestamp, 0).UTC().Format("2006-01-02")
credentialScope := fmt.Sprintf("%s/%s/tc3_request", date, service)
hashedCanonicalRequest := sha256hex(canonicalRequest)
string2sign := fmt.Sprintf("%s\n%d\n%s\n%s",
algorithm,
```

```
timestamp,
credentialScope,
hashedCanonicalRequest)
fmt.Println(string2sign)

// step 3: sign string
secretDate := hmacsha256(date, "TC3"+secretKey)
secretService := hmacsha256(service, secretDate)
secretSigning := hmacsha256("tc3_request", secretService)
signature := hex.EncodeToString([]byte(hmacsha256(string2sign, secretSigning)))
fmt.Println(signature)

// step 4: build authorization
authorization := fmt.Sprintf("%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm,
secretId,
credentialScope,
signedHeaders,
signature)
fmt.Println(authorization)

curl := fmt.Sprintf(`curl -X POST https://%s\
-H "Authorization: %s"\
-H "Content-Type: application/json; charset=utf-8"\
-H "Host: %s" -H "X-TC-Action: %s"\
-H "X-TC-Timestamp: %d"\
-H "X-TC-Version: %s"\
-H "X-TC-Region: %s"\
-d '%s'`, host, authorization, host, action, timestamp, version, region, payload)
fmt.Println(curl)
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$host = "cvm.tencentcloudapi.com";
$service = "cvm";
$version = "2017-03-12";
$action = "DescribeInstances";
$region = "ap-guangzhou";
// $timestamp = time();
$timestamp = 1551113065;
$algorithm = "TC3-HMAC-SHA256";

// step 1: build canonical request string
$httpRequestMethod = "POST";
$canonicalUri = "/";
$canonicalQueryString = "";
$canonicalHeaders = implode("\n", [
"content-type:application/json; charset=utf-8",
"host:".$host,
"x-tc-action:".strtolower($action),
```

```
""
]);
$signedHeaders = implode(";", [
"content-type",
"host",
"x-tc-action",
]);
$payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}';
$hashedRequestPayload = hash("SHA256", $payload);
$canonicalRequest = $httpRequestMethod."\n"
.$canonicalUri."\n"
.$canonicalQueryString."\n"
.$canonicalHeaders."\n"
.$signedHeaders."\n"
.$hashedRequestPayload;
echo $canonicalRequest.PHP_EOL;

// step 2: build string to sign
$date = gmdate("Y-m-d", $timestamp);
$credentialScope = $date."/".$service."/tc3_request";
$hashedCanonicalRequest = hash("SHA256", $canonicalRequest);
$stringToSign = $algorithm."\n"
.$timestamp."\n"
.$credentialScope."\n"
.$hashedCanonicalRequest;
echo $stringToSign.PHP_EOL;

// step 3: sign string
$secretDate = hash_hmac("SHA256", $date, "TC3".$secretKey, true);
$secretService = hash_hmac("SHA256", $service, $secretDate, true);
$secretSigning = hash_hmac("SHA256", "tc3_request", $secretService, true);
$signature = hash_hmac("SHA256", $stringToSign, $secretSigning);
echo $signature.PHP_EOL;

// step 4: build authorization
$authorization = $algorithm
." Credential=".$secretId."/".$credentialScope
.", SignedHeaders=".$signedHeaders.", Signature=".$signature;
echo $authorization.PHP_EOL;

$curl = "curl -X POST https://".$host
.' -H "Authorization: '.$authorization.'"'
.' -H "Content-Type: application/json; charset=utf-8"'
.' -H "Host: '.$host.'"'
.' -H "X-TC-Action: '.$action.'"'
.' -H "X-TC-Timestamp: '.$timestamp.'"'
.' -H "X-TC-Version: '.$version.'"'
.' -H "X-TC-Region: '.$region.'"'
." -d '".$payload."'";
echo $curl.PHP_EOL;
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'digest'
```

```
require 'json'
require 'time'
require 'openssl'

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

service = 'cvm'
host = 'cvm.tencentcloudapi.com'
endpoint = 'https://' + host
region = 'ap-guangzhou'
action = 'DescribeInstances'
version = '2017-03-12'
algorithm = 'TC3-HMAC-SHA256'
# timestamp = Time.now.to_i
timestamp = 1551113065
date = Time.at(timestamp).utc.strftime('%Y-%m-%d')

# ************* 步骤 1：拼接规范请求串 *************
http_request_method = 'POST'
canonical_uri = '/'
canonical_querystring = ''
canonical_headers = "content-type:application/json; charset=utf-8\nhost:#{host}\nx-tc-action:#{action.downcase}\n"
signed_headers = 'content-type;host;x-tc-action'
# params = { 'Limit' => 1, 'Filters' => [{ 'Name' => 'instance-name', 'Values' => ['未命名'] }] }
# payload = JSON.generate(params, { 'ascii_only' => true, 'space' => ' ' })
# json will generate in random order, to get specified result in example, we hard-code it here.
payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
hashed_request_payload = Digest::SHA256.hexdigest(payload)
canonical_request = [
http_request_method,
canonical_uri,
canonical_querystring,
canonical_headers,
signed_headers,
hashed_request_payload,
].join("\n")

puts canonical_request

# ************* 步骤 2：拼接待签名字符串 *************
credential_scope = date + '/' + service + '/' + 'tc3_request'
hashed_request_payload = Digest::SHA256.hexdigest(canonical_request)
string_to_sign = [
algorithm,
timestamp.to_s,
credential_scope,
hashed_request_payload,
].join("\n")
puts string_to_sign

# ************* 步骤 3：计算签名 *************
digest = OpenSSL::Digest.new('sha256')
secret_date = OpenSSL::HMAC.digest(digest, 'TC3' + secret_key, date)
```

```
secret_service = OpenSSL::HMAC.digest(digest, secret_date, service)
secret_signing = OpenSSL::HMAC.digest(digest, secret_service, 'tc3_request')
signature = OpenSSL::HMAC.hexdigest(digest, secret_signing, string_to_sign)
puts signature

# ************* 步骤 4：拼接 Authorization *************
authorization = "#{algorithm} Credential=#{secret_id}/#{credential_scope}, SignedHeaders=#{signed_headers}, Signature=#{si
gnature}"
puts authorization

puts 'curl -X POST ' + endpoint \
+ ' -H "Authorization: ' + authorization + '"' \
+ ' -H "Content-Type: application/json; charset=utf-8"' \
+ ' -H "Host: ' + host + '"' \
+ ' -H "X-TC-Action: ' + action + '"' \
+ ' -H "X-TC-Timestamp: ' + timestamp.to_s + '"' \
+ ' -H "X-TC-Version: ' + version + '"' \
+ ' -H "X-TC-Region: ' + region + '"' \
+ " -d '" + payload + "'"
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Security.Cryptography;
using System.Text;

public class Application
{
public static string SHA256Hex(string s)
{
using (SHA256 algo = SHA256.Create())
{
byte[] hashbytes = algo.ComputeHash(Encoding.UTF8.GetBytes(s));
StringBuilder builder = new StringBuilder();
for (int i = 0; i < hashbytes.Length; ++i)
{
builder.Append(hashbytes[i].ToString("x2"));
}
return builder.ToString();
}
}

public static byte[] HmacSHA256(byte[] key, byte[] msg)
{
using (HMACSHA256 mac = new HMACSHA256(key))
{
return mac.ComputeHash(msg);
}
}

public static Dictionary<String, String> BuildHeaders(string secretid,
string secretkey, string service, string endpoint, string region,
string action, string version, DateTime date, string requestPayload)
{
string datestr = date.ToString("yyyy-MM-dd");
```

```
DateTime startTime = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc);
long requestTimestamp = (long)Math.Round((date - startTime).TotalMilliseconds, MidpointRounding.AwayFromZero) / 1000;
// ************* 步骤 1: 拼接规范请求串 *************
string algorithm = "TC3-HMAC-SHA256";
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string contentType = "application/json";
string canonicalHeaders = "content-type:" + contentType + "; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.ToLower() + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string hashedRequestPayload = SHA256Hex(requestPayload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
Console.WriteLine(canonicalRequest);

// ************* 步骤 2: 拼接待签名字符串 *************
string credentialScope = datestr + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = SHA256Hex(canonicalRequest);
string stringToSign = algorithm + "\n"
+ requestTimestamp.ToString() + "\n"
+ credentialScope + "\n"
+ hashedCanonicalRequest;
Console.WriteLine(stringToSign);

// ************* 步骤 3: 计算签名 *************
byte[] tc3SecretKey = Encoding.UTF8.GetBytes("TC3" + secretkey);
byte[] secretDate = HmacSHA256(tc3SecretKey, Encoding.UTF8.GetBytes(datestr));
byte[] secretService = HmacSHA256(secretDate, Encoding.UTF8.GetBytes(service));
byte[] secretSigning = HmacSHA256(secretService, Encoding.UTF8.GetBytes("tc3_request"));
byte[] signatureBytes = HmacSHA256(secretSigning, Encoding.UTF8.GetBytes(stringToSign));
string signature = BitConverter.ToString(signatureBytes).Replace("-", "").ToLower();
Console.WriteLine(signature);

// ************* 步骤 4: 拼接 Authorization *************
string authorization = algorithm + " "
+ "Credential=" + secretid + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", "
+ "Signature=" + signature;
Console.WriteLine(authorization);

Dictionary<string, string> headers = new Dictionary<string, string>();
headers.Add("Authorization", authorization);
headers.Add("Host", endpoint);
headers.Add("Content-Type", contentType + "; charset=utf-8");
headers.Add("X-TC-Timestamp", requestTimestamp.ToString());
headers.Add("X-TC-Version", version);
headers.Add("X-TC-Action", action);
headers.Add("X-TC-Region", region);
return headers;
}
public static void Main(string[] args)
```

```
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";

// 此处由于示例规范的原因，采用时间戳2019-02-26 00:44:25，此参数作为示例，如果在项目中，您应当使用：
// DateTime date = DateTime.UtcNow;
// 注意时区，建议此时间统一采用UTC时间戳，否则容易出错
DateTime date = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc).AddSeconds(1551113065);
string requestPayload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name
\"}]}";

Dictionary<string, string> headers = BuildHeaders(SECRET_ID, SECRET_KEY, service
, endpoint, region, action, version, date, requestPayload);

Console.WriteLine("POST https://cvm.tencentcloudapi.com");
foreach (KeyValuePair<string, string> kv in headers)
{
Console.WriteLine(kv.Key + ": " + kv.Value);
}
Console.WriteLine();
Console.WriteLine(requestPayload);
}
}
```

NodeJS

```
const crypto = require('crypto');

function sha256(message, secret = '', encoding) {
const hmac = crypto.createHmac('sha256', secret)
return hmac.update(message).digest(encoding)
}

function getHash(message, encoding = 'hex') {
const hash = crypto.createHash('sha256')
return hash.update(message).digest(encoding)
}

function getDate(timestamp) {
const date = new Date(timestamp * 1000)
const year = date.getUTCFullYear()
const month = ('0' + (date.getUTCMonth() + 1)).slice(-2)
const day = ('0' + date.getUTCDate()).slice(-2)
return `${year}-${month}-${day}`
}

function main(){
```

```
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const service = "cvm"
const region = "ap-guangzhou"
const action = "DescribeInstances"
const version = "2017-03-12"
//const timestamp = getTime()
const timestamp = 1551113065
//时间处理，获取世界时间日期
const date = getDate(timestamp)

// ************* 步骤 1：拼接规范请求串 *************
const payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}"

const hashedRequestPayload = getHash(payload);
const httpRequestMethod = "POST"
const canonicalUri = "/"
const canonicalQueryString = ""
const canonicalHeaders = "content-type:application/json; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.toLowerCase() + "\n"
const signedHeaders = "content-type;host;x-tc-action"

const canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload
console.log(canonicalRequest)

// ************* 步骤 2：拼接待签名字符串 *************
const algorithm = "TC3-HMAC-SHA256"
const hashedCanonicalRequest = getHash(canonicalRequest);
const credentialScope = date + "/" + service + "/" + "tc3_request"
const stringToSign = algorithm + "\n" +
timestamp + "\n" +
credentialScope + "\n" +
hashedCanonicalRequest
console.log(stringToSign)

// ************* 步骤 3：计算签名 *************
const kDate = sha256(date, 'TC3' + SECRET_KEY)
const kService = sha256(service, kDate)
const kSigning = sha256('tc3_request', kService)
const signature = sha256(stringToSign, kSigning, 'hex')
console.log(signature)

// ************* 步骤 4：拼接 Authorization *************
const authorization = algorithm + " " +
"Credential=" + SECRET_ID + "/" + credentialScope + ", " +
"SignedHeaders=" + signedHeaders + ", " +
```

```
"Signature=" + signature
console.log(authorization)

const curlcmd = 'curl -X POST ' + "https://" + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + endpoint + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + timestamp.toString() + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + "'"
console.log(curlcmd)
}
main()
```

C++

```
#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <iomanip>
#include <sstream>
#include <string>
#include <stdio.h>
#include <time.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

using namespace std;

string get_data(int64_t &timestamp)
{
string utcDate;
char buff[20] = {0};
// time_t timenow;
struct tm sttime;
sttime = *gmtime(&timestamp);
strftime(buff, sizeof(buff), "%Y-%m-%d", &sttime);
utcDate = string(buff);
return utcDate;
}

string int2str(int64_t n)
{
std::stringstream ss;
ss << n;
return ss.str();
}

string sha256Hex(const string &str)
{
char buf[3];
unsigned char hash[SHA256_DIGEST_LENGTH];
SHA256_CTX sha256;
SHA256_Init(&sha256);
```

```
SHA256_Update(&sha256, str.c_str(), str.size());
SHA256_Final(hash, &sha256);
std::string NewString = "";
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
snprintf(buf, sizeof(buf), "%02x", hash[i]);
NewString = NewString + buf;
}
return NewString;
}

string HmacSha256(const string &key, const string &input)
{
unsigned char hash[32];

HMAC_CTX *h;
#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX hmac;
HMAC_CTX_init(&hmac);
h = &hmac;
#else
h = HMAC_CTX_new();
#endif

HMAC_Init_ex(h, &key[0], key.length(), EVP_sha256(), NULL);
HMAC_Update(h, ( unsigned char* )&input[0], input.length());
unsigned int len = 32;
HMAC_Final(h, hash, &len);

#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX_cleanup(h);
#else
HMAC_CTX_free(h);
#endif

std::stringstream ss;
ss << std::setfill('0');
for (int i = 0; i < len; i++)
{
ss << hash[i];
}

return (ss.str());
}

string HexEncode(const string &input)
{
static const char* const lut = "0123456789abcdef";
size_t len = input.length();

string output;
output.reserve(2 * len);
for (size_t i = 0; i < len; ++i)
{
const unsigned char c = input[i];
output.push_back(lut[c >> 4]);
output.push_back(lut[c & 15]);
```

```
}
return output;
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string host = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";
int64_t timestamp = 1551113065;
string date = get_data(timestamp);

// ************* 步骤 1: 拼接规范请求串 *************
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string lower = action;
std::transform(action.begin(), action.end(), lower.begin(), ::tolower);
string canonicalHeaders = string("content-type:application/json; charset=utf-8\n")
+ "host:" + host + "\n"
+ "x-tc-action:" + lower + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}";
string hashedRequestPayload = sha256Hex(payload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
cout << canonicalRequest << endl;

// ************* 步骤 2: 拼接待签名字符串 *************
string algorithm = "TC3-HMAC-SHA256";
string RequestTimestamp = int2str(timestamp);
string credentialScope = date + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = sha256Hex(canonicalRequest);
string stringToSign = algorithm + "\n" + RequestTimestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
cout << stringToSign << endl;

// ************* 步骤 3: 计算签名 **************
string kKey = "TC3" + SECRET_KEY;
string kDate = HmacSha256(kKey, date);
string kService = HmacSha256(kDate, service);
string kSigning = HmacSha256(kService, "tc3_request");
string signature = HexEncode(HmacSha256(kSigning, stringToSign));
cout << signature << endl;

// ************* 步骤 4: 拼接 Authorization *************
```

```
string authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
cout << authorization << endl;

string curlcmd = "curl -X POST https://" + host + "\n"
+ " -H \"Authorization: " + authorization + "\"\n"
+ " -H \"Content-Type: application/json; charset=utf-8\"" + "\n"
+ " -H \"Host: " + host + "\"\n"
+ " -H \"X-TC-Action: " + action + "\"\n"
+ " -H \"X-TC-Timestamp: " + RequestTimestamp + "\"\n"
+ " -H \"X-TC-Version: " + version + "\"\n"
+ " -H \"X-TC-Region: " + region + "\"\n"
+ " -d '" + payload + "\'";
cout << curlcmd << endl;
return 0;
};
```

C

```
#include <ctype.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <stdint.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

void get_utc_date(int64_t timestamp, char* utc, int len)
{
// time_t timenow;
struct tm sttime;
sttime = *gmtime(&timestamp);
strftime(utc, len, "%Y-%m-%d", &sttime);
}

void sha256_hex(const char* str, char* result)
{
char buf[3];
unsigned char hash[SHA256_DIGEST_LENGTH];
SHA256_CTX sha256;
SHA256_Init(&sha256);
SHA256_Update(&sha256, str, strlen(str));
SHA256_Final(hash, &sha256);
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
snprintf(buf, sizeof(buf), "%02x", hash[i]);
strcat(result, buf);
}
}

void hmac_sha256(const char* key, int key_len,
const char* input, int input_len,
unsigned char* output, unsigned int* output_len)
{
HMAC_CTX *h;
```

```
#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX hmac;
HMAC_CTX_init(&hmac);
h = &hmac;
#else
h = HMAC_CTX_new();
#endif

HMAC_Init_ex(h, key, key_len, EVP_sha256(), NULL);
HMAC_Update(h, ( unsigned char* )input, input_len);
HMAC_Final(h, output, output_len);


#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX_cleanup(h);
#else
HMAC_CTX_free(h);
#endif

}

void hex_encode(const char* input, int input_len, char* output)
{
static const char* const lut = "0123456789abcdef";

char add_out[128] = {0};
char temp[2] = {0};
for (size_t i = 0; i < input_len; ++i)
{
const unsigned char c = input[i];
temp[0] = lut[c >> 4];
strcat(add_out, temp);
temp[0] = lut[c & 15];
strcat(add_out, temp);
}
strncpy(output, add_out, 128);
}

void lowercase(const char * src, char * dst)
{
for (int i = 0; src[i]; i++)
{
dst[i] = tolower(src[i]);
}
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const char* SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const char* SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");
const char* service = "cvm";
const char* host = "cvm.tencentcloudapi.com";
const char* region = "ap-guangzhou";
const char* action = "DescribeInstances";
```

```
const char* version = "2017-03-12";
int64_t timestamp = 1551113065;
char date[20] = {0};
get_utc_date(timestamp, date, sizeof(date));

// ************* 步骤 1: 拼接规范请求串 *************
const char* http_request_method = "POST";
const char* canonical_uri = "/";
const char* canonical_query_string = "";
char canonical_headers[100] = {"content-type:application/json; charset=utf-8\nhost:"};
strcat(canonical_headers, host);
strcat(canonical_headers, "\nx-tc-action:");
char value[100] = {0};
lowercase(action, value);
strcat(canonical_headers, value);
strcat(canonical_headers, "\n");
const char* signed_headers = "content-type;host;x-tc-action";
const char* payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name
\"}]}";
char hashed_request_payload[100] = {0};
sha256_hex(payload, hashed_request_payload);

char canonical_request[256] = {0};
sprintf(canonical_request, "%s\n%s\n%s\n%s\n%s\n%s", http_request_method,
canonical_uri, canonical_query_string, canonical_headers,
signed_headers, hashed_request_payload);
printf("%s\n", canonical_request);


// ************* 步骤 2: 拼接待签名字符串 *************
const char* algorithm = "TC3-HMAC-SHA256";
char request_timestamp[16] = {0};
sprintf(request_timestamp, "%d", timestamp);
char credential_scope[64] = {0};
strcat(credential_scope, date);
sprintf(credential_scope, "%s/%s/tc3_request", date, service);
char hashed_canonical_request[100] = {0};
sha256_hex(canonical_request, hashed_canonical_request);
char string_to_sign[256] = {0};
sprintf(string_to_sign, "%s\n%s\n%s\n%s", algorithm, request_timestamp,
credential_scope, hashed_canonical_request);
printf("%s\n", string_to_sign);

// ************* 步骤 3: 计算签名 ***************
char k_key[64] = {0};
sprintf(k_key, "%s%s", "TC3", SECRET_KEY);
unsigned char k_date[64] = {0};
unsigned int output_len = 0;
hmac_sha256(k_key, strlen(k_key), date, strlen(date), k_date, &output_len);
unsigned char k_service[64] = {0};
hmac_sha256(k_date, output_len, service, strlen(service), k_service, &output_len);
unsigned char k_signing[64] = {0};
hmac_sha256(k_service, output_len, "tc3_request", strlen("tc3_request"), k_signing, &output_len);
unsigned char k_hmac_sha_sign[64] = {0};
hmac_sha256(k_signing, output_len, string_to_sign, strlen(string_to_sign), k_hmac_sha_sign, &output_len);

char signature[128] = {0};
```

```
hex_encode(k_hmac_sha_sign, output_len, signature);
printf("%s\n", signature);

// ************* 步骤 4: 拼接 Authorization *************
char authorization[512] = {0};
sprintf(authorization, "%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm, SECRET_ID, credential_scope, signed_headers, signature);
printf("%s\n", authorization);

char curlcmd[10240] = {0};
sprintf(curlcmd, "curl -X POST https://%s\n \
-H \"Authorization: %s\"\n \
-H \"Content-Type: application/json; charset=utf-8\"\n \
-H \"Host: %s\"\n \
-H \"X-TC-Action: %s\"\n \
-H \"X-TC-Timestamp: %s\"\n \
-H \"X-TC-Version: %s\"\n \
-H \"X-TC-Region: %s\"\n \
-d \'%s\'",
host, authorization, host, action, request_timestamp, version, region, payload);
printf("%s\n", curlcmd);
return 0;
}
```

其他语言

- Lua: GitHub
- Swift: GitHub
- Dart: GitHub
- Shell(Bash): GitHub

签名失败

存在以下签名失败的错误码，请根据实际情况处理。

错误码	错误描述
AuthFailure.SignatureExpire	签名过期。Timestamp 与服务器接收到请求的时间相差不得超过五分钟。
AuthFailure.SecretIdNotFound	密钥不存在。请到控制台查看密钥是否被禁用，是否少复制了字符或者多了字符。
AuthFailure.SignatureFailure	签名错误。可能是签名计算错误，或者签名与实际发送的内容不相符合，也有可能是密钥 SecretKey 错误导致的。
AuthFailure.TokenFailure	临时证书 Token 错误。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。

签名方法

最近更新时间：2024-12-25 01:31:55

签名方法 v1 简单易用，但是功能和安全性都不如签名方法 v3，推荐使用签名方法 v3。

首次接触，建议使用 API Explorer 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v1”，可以生成签名过程进行验证，并提供了部分编程语言的签名示例，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 Python、Java、PHP、Go、NodeJS、.NET、C++、Ruby。

推荐使用 API Explorer　　点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（Signature）以验证请求者身份。
签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往 云API密钥页面 申请，否则无法调用云 API 接口。

1. 申请安全凭证

在第一次使用云 API 之前，请前往 云 API 密钥页面 申请安全凭证。
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- **用户必须严格保管安全凭证，避免泄露。**

申请安全凭证的具体步骤如下：

1. 登录 腾讯云管理中心控制台 。
2. 前往 云 API 密钥 的控制台页面
3. 在 云 API 密钥 页面，单击【新建密钥】即可以创建一对 SecretId/SecretKey。

注意：每个账号最多可以拥有两对 SecretId/SecretKey。

2. 生成签名串

有了安全凭证 SecretId 和 SecretKey 后，就可以生成签名串了。以下是使用签名方法 v1 生成签名串的详细过程：

假设用户的 SecretId 和 SecretKey 分别是：

- SecretId: `AKID********************************`
- SecretKey: `********************************`

注意：这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作！

以云服务器查看实例列表（DescribeInstances）请求为例，当用户调用这一接口时，其请求参数可能如下：

参数名称	中文	参数值
Action	方法名	DescribeInstances
SecretId	密钥 ID	`AKID********************************`
Timestamp	当前时间戳	1465185768
Nonce	随机正整数	11886
Region	实例所在区域	ap-guangzhou
InstanceIds.0	待查询的实例 ID	ins-09dx96dg
Offset	偏移量	0
Limit	最大允许输出	20
Version	接口版本号	2017-03-12

这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数。

2.1. 对参数排序

首先对所有请求参数按参数名的字典序（ ASCII 码）升序排序。注意：1）只按参数名进行排序，参数值保持对应即可，不参与比大小；2）按 ASCII 码比大小，如 InstanceIds.2 要排在 InstanceIds.12 后面，不是按字母表，也不是按数值。用户可以借助编程语言中的相关排序函数来实现这一功能，如 PHP 中的 ksort 函数。上述示例参数的排序结果如下：

```
{
'Action' : 'DescribeInstances',
'InstanceIds.0' : 'ins-09dx96dg',
'Limit' : 20,
'Nonce' : 11886,
'Offset' : 0,
'Region' : 'ap-guangzhou',
'SecretId' : 'AKID********************************',
'Timestamp' : 1465185768,
'Version': '2017-03-12',
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

2.2. 拼接请求字符串

此步骤生成请求字符串。
将把上一步排序好的请求参数格式化成“参数名称=参数值”的形式，如对 Action 参数，其参数名称为 "Action"，参数值为 "DescribeInstances"，因此格式化后就为 Action=DescribeInstances 。
注意：“参数值”为原始值而非 url 编码后的值。

然后将格式化后的各个参数用"&"拼接在一起，最终生成的请求字符串为：

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*******
************************&Timestamp=1465185768&Version=2017-03-12
```

2.3. 拼接签名原文字符串

此步骤生成签名原文字符串。
签名原文字符串由以下几个参数构成：

1. 请求方法: 支持 POST 和 GET 方式，这里使用 GET 请求，注意方法为全大写。
2. 请求主机:查看实例列表(DescribeInstances)的请求域名为：cvm.tencentcloudapi.com。实际的请求域名根据接口所属模块的不同而不同，详见各接口说明。
3. 请求路径: 当前版本云API的请求路径固定为 / 。
4. 请求字符串: 即上一步生成的请求字符串。

签名原文串的拼接规则为： 请求方法 + 请求主机 +请求路径 + ? + 请求字符串 。

示例的拼接结果为：

```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-gu
angzhou&SecretId=AKID*******************************&Timestamp=1465185768&Version=2017-03-12
```

2.4. 生成签名串

此步骤生成签名串。
首先使用 HMAC-SHA1 算法对上一步中获得的**签名原文字符串**进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。

具体代码如下，以 PHP 语言为例：

```
$secretKey = '********************************';
$srcStr = 'GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&R
egion=ap-guangzhou&SecretId=AKID*******************************&Timestamp=1465185768&Version=2017-03-12';
```

```
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

最终得到的签名串为：

```
7RAM2xfNMO9EiVTNmPg06MRnCvQ=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

3. 签名串编码

生成的签名串并不能直接作为请求参数，需要对其进行 URL 编码。

如上一步生成的签名串为 7RAM2xfNMO9EiVTNmPg06MRnCvQ= ，最终得到的签名串请求参数（Signature）为：7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D，它将用于生成最终的请求 URL。

注意：如果用户的请求方法是 GET，或者请求方法为 POST 同时 Content-Type 为 application/x-www-form-urlencoded，则发送请求时所有请求参数的值均需要做 URL 编码，参数键和=符号不需要编码。非 ASCII 字符在 URL 编码前需要先以 UTF-8 进行编码。

注意：有些编程语言的网络库会自动为所有参数进行 urlencode，在这种情况下，就不需要对签名串进行 URL 编码了，否则两次 URL 编码会导致签名失败。

注意：其他参数值也需要进行编码，编码采用 RFC 3986。使用 %XY 对特殊字符例如汉字进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F），使用小写将引发错误。

4. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理。

错误代码	错误描述
AuthFailure.SignatureExpire	签名过期
AuthFailure.SecretIdNotFound	密钥不存在
AuthFailure.SignatureFailure	签名错误
AuthFailure.TokenFailure	token 错误
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）

5. 签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0 ，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 SDK 中心。当前支持的编程语言有：

- Python
- Java
- PHP
- Go
- NodeJS
- .NET
- C++
- Ruby

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- Signature Demo

为了更清楚的解释签名过程，下面以实际编程语言为例，将上述的签名过程具体实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK 。

最终输出的 url 可能为： `https://cvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-`

guangzhou&SecretId=AKID********************************&Signature=7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D&Timestamp=1465185768&Version=2017-03-12 。

注意：由于示例中的密钥是虚构的，时间戳也不是系统当前时间，因此如果将此 url 在浏览器中打开或者用 curl 等命令调用时会返回鉴权错误：签名过期。为了得到一个可以正常返回的 url ，需要修改示例中的 SecretId 和 SecretKey 为真实的密钥，并使用系统当前时间戳作为 Timestamp 。

注意：在下面的示例中，不同编程语言，甚至同一语言每次执行得到的 url 可能都有所不同，表现为参数的顺序不同，但这并不影响正确性。只要所有参数都在，且签名计算正确即可。

注意：以下代码仅适用于 API 3.0，不能直接用于其他的签名流程，请以对应的实际文档为准。

Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPIDemo {
private final static String CHARSET = "UTF-8";

public static String sign(String s, String key, String method) throws Exception {
Mac mac = Mac.getInstance(method);
SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
mac.init(secretKeySpec);
byte[] hash = mac.doFinal(s.getBytes(CHARSET));
return DatatypeConverter.printBase64Binary(hash);
}

public static String getStringToSign(TreeMap<String, Object> params) {
StringBuilder s2s = new StringBuilder("GETcvm.tencentcloudapi.com/?");
// 签名时要求对参数进行字典排序，此处用TreeMap保证顺序
for (String k : params.keySet()) {
s2s.append(k).append("=").append(params.get(k).toString()).append("&");
}
return s2s.toString().substring(0, s2s.length() - 1);
}

public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException {
StringBuilder url = new StringBuilder("https://cvm.tencentcloudapi.com/?");
// 实际请求的url中对参数顺序没有要求
for (String k : params.keySet()) {
// 需要对请求串进行urlencode，由于key都是英文字母，故此处仅对其value进行urlencode
url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).append("&");
}
return url.toString().substring(0, url.length() - 1);
}

public static void main(String[] args) throws Exception {
TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap可以自动排序
// 实际调用时应当使用随机数，例如：params.put("Nonce", new Random().nextInt(java.lang.Integer.MAX_VALUE));
params.put("Nonce", 11886); // 公共参数
// 实际调用时应当使用系统当前时间，例如： params.put("Timestamp", System.currentTimeMillis() / 1000);
params.put("Timestamp", 1465185768); // 公共参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
params.put("SecretId", System.getenv("TENCENTCLOUD_SECRET_ID")); // 公共参数
```

```
params.put("Action", "DescribeInstances"); // 公共参数
params.put("Version", "2017-03-12"); // 公共参数
params.put("Region", "ap-guangzhou"); // 公共参数
params.put("Limit", 20); // 业务参数
params.put("Offset", 0); // 业务参数
params.put("InstanceIds.0", "ins-09dx96dg"); // 业务参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
params.put("Signature", sign(getStringToSign(params), System.getenv("TENCENTCLOUD_SECRET_KEY"), "HmacSHA1")); // 公共参数
System.out.println(getUrl(params));
}
}
```

Python

注意：如果是在 Python 2 环境中运行，需要先安装 requests 依赖包： `pip install requests` 。

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import os
import time

import requests

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

def get_string_to_sign(method, endpoint, params):
s = method + endpoint + "/?"
query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
return s + query_str

def sign_str(key, s, method):
hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
return base64.b64encode(hmac_str)

if __name__ == '__main__':
endpoint = "cvm.tencentcloudapi.com"
data = {
'Action' : 'DescribeInstances',
'InstanceIds.0' : 'ins-09dx96dg',
'Limit' : 20,
'Nonce' : 11886,
'Offset' : 0,
'Region' : 'ap-guangzhou',
'SecretId' : secret_id,
'Timestamp' : 1465185768, # int(time.time())
'Version': '2017-03-12'
}
s = get_string_to_sign("GET", endpoint, data)
data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
print(data["Signature"])
# 此处会实际调用，成功后可能产生计费
```

```
# resp = requests.get("https://" + endpoint, params=data)
# print(resp.url)
```

Golang

```
package main

import (
"bytes"
"crypto/hmac"
"crypto/sha1"
"encoding/base64"
"fmt"
"os"
"sort"
"strconv"
)

func main() {
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID******************************
secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ******************************
secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
params := map[string]string{
"Nonce": "11886",
"Timestamp": strconv.Itoa(1465185768),
"Region": "ap-guangzhou",
"SecretId": secretId,
"Version": "2017-03-12",
"Action": "DescribeInstances",
"InstanceIds.0": "ins-09dx96dg",
"Limit": strconv.Itoa(20),
"Offset": strconv.Itoa(0),
}

var buf bytes.Buffer
buf.WriteString("GET")
buf.WriteString("cvm.tencentcloudapi.com")
buf.WriteString("/")
buf.WriteString("?")

// sort keys by ascii asc order
keys := make([]string, 0, len(params))
for k, _ := range params {
keys = append(keys, k)
}
sort.Strings(keys)

for i := range keys {
k := keys[i]
buf.WriteString(k)
buf.WriteString("=")
buf.WriteString(params[k])
buf.WriteString("&")
}
buf.Truncate(buf.Len() - 1)
```

```
hashed := hmac.New(sha1.New, []byte(secretKey))
hashed.Write(buf.Bytes())

fmt.Println(base64.StdEncoding.EncodeToString(hashed.Sum(nil)))
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$param["Nonce"] = 11886;//rand();
$param["Timestamp"] = 1465185768;//time();
$param["Region"] = "ap-guangzhou";
$param["SecretId"] = $secretId;
$param["Version"] = "2017-03-12";
$param["Action"] = "DescribeInstances";
$param["InstanceIds.0"] = "ins-09dx96dg";
$param["Limit"] = 20;
$param["Offset"] = 0;

ksort($param);

$signStr = "GETcvm.tencentcloudapi.com/?";
foreach ( $param as $key => $value ) {
$signStr = $signStr . $key . "=" . $value . "&";
}
$signStr = substr($signStr, 0, -1);

$signature = base64_encode(hash_hmac("sha1", $signStr, $secretKey, true));
echo $signature.PHP_EOL;
// need to install and enable curl extension in php.ini
// $param["Signature"] = $signature;
// $url = "https://cvm.tencentcloudapi.com/?".http_build_query($param);
// echo $url.PHP_EOL;
// $ch = curl_init();
// curl_setopt($ch, CURLOPT_URL, $url);
// $output = curl_exec($ch);
// curl_close($ch);
// echo json_decode($output);
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'time'
require 'openssl'
require 'base64'

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
```

```
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

method = 'GET'
endpoint = 'cvm.tencentcloudapi.com'
data = {
'Action' => 'DescribeInstances',
'InstanceIds.0' => 'ins-09dx96dg',
'Limit' => 20,
'Nonce' => 11886,
'Offset' => 0,
'Region' => 'ap-guangzhou',
'SecretId' => secret_id,
'Timestamp' => 1465185768, # Time.now.to_i
'Version' => '2017-03-12',
}
sign = method + endpoint + '/?'
params = []
data.sort.each do |item|
params << "#{item[0]}=#{item[1]}"
end
sign += params.join('&')
digest = OpenSSL::Digest.new('sha1')
data['Signature'] = Base64.encode64(OpenSSL::HMAC.digest(digest, secret_key, sign))
puts data['Signature']

# require 'net/http'
# uri = URI('https://' + endpoint)
# uri.query = URI.encode_www_form(data)
# p uri
# res = Net::HTTP.get_response(uri)
# puts res.body
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Security.Cryptography;
using System.Text;

public class Application {
public static string Sign(string signKey, string secret)
{
string signRet = string.Empty;
using (HMACSHA1 mac = new HMACSHA1(Encoding.UTF8.GetBytes(signKey)))
{
byte[] hash = mac.ComputeHash(Encoding.UTF8.GetBytes(secret));
signRet = Convert.ToBase64String(hash);
}
return signRet;
}
public static string MakeSignPlainText(SortedDictionary<string, string> requestParams, string requestMethod, string reques
tHost, string requestPath)
{
string retStr = "";
retStr += requestMethod;
```

```
retStr += requestHost;
retStr += requestPath;
retStr += "?";
string v = "";
foreach (string key in requestParams.Keys)
{
v += string.Format("{0}={1}&", key, requestParams[key]);
}
retStr += v.TrimEnd('&');
return retStr;
}

public static void Main(string[] args)
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";
double RequestTimestamp = 1465185768; // 时间戳 2019-02-26 00:44:25,此参数作为示例，以实际为准
// long timestamp = ToTimestamp() / 1000;
// string requestTimestamp = timestamp.ToString();
Dictionary<string, string> param = new Dictionary<string, string>();
param.Add("Limit", "20");
param.Add("Offset", "0");
param.Add("InstanceIds.0", "ins-09dx96dg");
param.Add("Action", action);
param.Add("Nonce", "11886");
// param.Add("Nonce", Math.Abs(new Random().Next()).ToString());

param.Add("Timestamp", RequestTimestamp.ToString());
param.Add("Version", version);

param.Add("SecretId", SECRET_ID);
param.Add("Region", region);
SortedDictionary<string, string> headers = new SortedDictionary<string, string>(param, StringComparer.Ordinal);
string sigInParam = MakeSignPlainText(headers, "GET", endpoint, "/");
string sigOutParam = Sign(SECRET_KEY, sigInParam);
Console.WriteLine(sigOutParam);
}
}
```

NodeJS

```
const crypto = require('crypto');

function get_req_url(params, endpoint){
params['Signature'] = encodeURIComponent(params['Signature']);
const url_strParam = sort_params(params)
return "https://" + endpoint + "/?" + url_strParam.slice(1);
}
```

```
function formatSignString(reqMethod, endpoint, path, strParam){
let strSign = reqMethod + endpoint + path + "?" + strParam.slice(1);
return strSign;
}
function sha1(secretKey, strsign){
let signMethodMap = {'HmacSHA1': "sha1"};
let hmac = crypto.createHmac(signMethodMap['HmacSHA1'], secretKey || "");
return hmac.update(Buffer.from(strsign, 'utf8')).digest('base64')
}

function sort_params(params){
let strParam = "";
let keys = Object.keys(params);
keys.sort();
for (let k in keys) {
//k = k.replace(/_/g, '.');
strParam += ("&" + keys[k] + "=" + params[keys[k]]);
}
return strParam
}

function main(){
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const Region = "ap-guangzhou"
const Version = "2017-03-12"
const Action = "DescribeInstances"
const Timestamp = 1465185768 // 时间戳 2016-06-06 12:02:48，此参数作为示例，以实际为准
// const Timestamp = Math.round(Date.now() / 1000)
const Nonce = 11886 // 随机正整数
//const nonce = Math.round(Math.random() * 65535)

let params = {};
params['Action'] = Action;
params['InstanceIds.0'] = 'ins-09dx96dg';
params['Limit'] = 20;
params['Offset'] = 0;
params['Nonce'] = Nonce;
params['Region'] = Region;
params['SecretId'] = SECRET_ID;
params['Timestamp'] = Timestamp;
params['Version'] = Version;

// 1. 对参数排序，并拼接请求字符串
strParam = sort_params(params)

// 2. 拼接签名原文字符串
const reqMethod = "GET";
const path = "/";
strSign = formatSignString(reqMethod, endpoint, path, strParam)
// console.log(strSign)
```

```
// 3. 生成签名串
params['Signature'] = sha1(SECRET_KEY, strSign)
console.log(params['Signature'])

// 4. 进行url编码并拼接请求url
// const req_url = get_req_url(params, endpoint)
// console.log(params['Signature'])
// console.log(req_url)
}
main()
```

返回结果

最近更新时间：2024-03-12 01:28:20

云 API 3.0 接口默认返回 JSON 数据，返回非 JSON 格式的接口会在文档中做出说明。返回 JSON 数据时最大限制为 50 MB，如果返回的数据超过最大限制，请求会失败并返回内部错误。请根据接口文档中给出的过滤功能（例如时间范围）或者分页功能，控制返回数据不要过大。

注意：目前只要请求被服务端正常处理了，响应的 HTTP 状态码均为200。例如返回的消息体里的错误码是签名失败，但 HTTP 状态码是200，而不是401。

正确返回结果

以云服务器的接口查看实例状态列表 (DescribeInstancesStatus) 2017-03-12 版本为例，若调用成功，其可能的返回如下为：

```
{
"Response": {
"TotalCount": 0,
"InstanceStatusSet": [],
"RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
}
}
```

- Response 及其内部的 RequestId 是固定的字段，无论请求成功与否，只要 API 处理了，则必定会返回。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 腾讯云客服 或 提交工单，并提供该 ID 来解决问题。
- 除了固定的字段外，其余均为具体接口定义的字段，不同的接口所返回的字段参见接口文档中的定义。此例中的 TotalCount 和 InstanceStatusSet 均为 DescribeInstancesStatus 接口定义的字段，由于调用请求的用户暂时还没有云服务器实例，因此 TotalCount 在此情况下的返回值为 0， InstanceStatusSet 列表为空。

错误返回结果

若调用失败，其返回值示例如下为：

```
{
"Response": {
"Error": {
"Code": "AuthFailure.SignatureFailure",
"Message": "The provided credentials could not be validated. Please check your signature is correct."
},
"RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
}
}
```

- Error 的出现代表着该请求调用失败。Error 字段连同其内部的 Code 和 Message 字段在调用失败时是必定返回的。
- Code 表示具体出错的错误码，当请求出错时可以先根据该错误码在公共错误码和当前接口对应的错误码列表里面查找对应原因和解决方案。
- Message 显示出了这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新，用户不应依赖这个返回值。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 腾讯云客服 或 提交工单，并提供该 ID 来解决问题。

公共错误码

返回结果中如果存在 Error 字段，则表示调用 API 接口失败。 Error 中的 Code 字段表示错误码，所有业务都可能出现的错误码为公共错误码。完整的错误码列表请参考本产品“API 文档”目录下的“错误码”页面。

参数类型

最近更新时间：2022-10-28 16:28:01

目前腾讯云 API 3.0 输入参数和输出参数支持如下几种数据格式：

- String: 字符串。
- Integer：整型，上限为无符号64位整数。SDK 3.0 不同编程语言支持的类型有所差异，建议以所使用编程语言的最大整型定义，例如 `Golang` 的 `uint64`。
- Boolean：布尔型。
- Float：浮点型。
- Double：双精度浮点型。
- Date：字符串，日期格式。例如：2022-01-01。
- Timestamp：字符串，时间格式。例如：2022-01-01 00:00:00。
- Timestamp ISO8601：ISO 8601 是由国际标准化组织（International Organization for Standardization，ISO）发布的关于日期和时间格式的国际标准，对应国标《GB/T 7408-2005数据元和交换格式信息交换日期和时间表示法》。建议以所使用编程语言的标准库进行格式解析。例如：2022-01-01T00:00:00+08:00。
- Binary：二进制内容，需要以特定协议请求和解析。

（新版）数据迁移相关接口
查询数据迁移任务列表

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询数据迁移任务列表

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrationJobs。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	数据迁移任务ID，如：dts-amm1jw5q 示例值：dts-amm1jw5q
JobName	否	String	数据迁移任务名称 示例值：dts-amm1jw5q
Status.N	否	Array of String	数据迁移任务状态，可取值包括：created(创建完成)、checking(校验中)、checkPass(校验通过)、checkNot 运行)、running(任务运行中)、readyComplete(准备完成)、success(任务成功)、failed(任务失败)、stopping 示例值：["running"]
SrcInstanceId	否	String	源实例ID，格式如：cdb-c1nl9rpv 示例值：cdb-c1nl9rpv
SrcRegion	否	String	源实例地域，如：ap-guangzhou 示例值：ap-guangzhou
SrcDatabaseType.N	否	Array of String	源实例数据库类型，如： sqlserver,mysql,mongodb,redis,tendis,keewidb,clickhouse,cynosdbmysql,percona,tdsqlpercon 示例值：["mysql"]
SrcAccessType.N	否	Array of String	源实例接入类型，值包括：extranet(外网)、vpncloud(云vpn接入的实例)、dcg(专线接入的实例)、ccn(云联网 cvm(cvm自建实例) 示例值：["cdb"]
DstInstanceId	否	String	目标实例ID，格式如：cdb-c1nl9rpv 示例值：cdb-c1nl9rpv
DstRegion	否	String	目标实例地域，如：ap-guangzhou 示例值：ap-guangzhou
DstDatabaseType.N	否	Array of String	目标源实例数据库类型，如： sqlserver,mysql,mongodb,redis,tendis,keewidb,clickhouse,cynosdbmysql,percona,tdsqlpercon 示例值：["mysql"]
DstAccessType.N	否	Array of String	目标实例接入类型，值包括：extranet(外网)、vpncloud(云vpn接入的实例)、dcg(专线接入的实例)、ccn(云联网 cvm(cvm自建实例) 示例值：["cdb"]

参数名称	必选	类型	描述
RunMode	否	String	任务运行模式，值包括：immediate(立即运行)，timed(定时运行) 示例值：immediate
OrderSeq	否	String	排序方式，可能取值为asc、desc，默认按照创建时间倒序 示例值：desc
Limit	否	Integer	返回实例数量，默认20，有效区间[1,100] 示例值：20
Offset	否	Integer	偏移量，默认为0 示例值：0
TagFilters.N	否	Array of TagFilter	标签过滤

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	迁移任务数量 示例值：1
JobList	Array of JobItem	迁移任务列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询数据迁移任务列表

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeMigrationJobs
<公共请求参数>

{
"JobId": "dts-amm1jw5q"
}
```

输出示例

```
{
"Response": {
"JobList": [
{
"Action": {
"AllAction": [
"view",
"modify",
"check",
"start",
"configure",
"stop",
"complete",
```

```
"resume",
"createCmpTask",
"isolate",
"offline",
"resize",
"recover"
],
"AllowedAction": [
"view",
"stop",
"complete",
"createCmpTask",
"view",
"isolate",
"resize"
]
},
"CompareTask": {
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk",
"Status": "consistent"
},
"CreateTime": "2022-07-11 16:20:49",
"BriefMsg": "",
"DumperResumeCtrl": "yes",
"AutoRetryTimeRangeMinutes": 5,
"StepInfo": {
"MasterSlaveDistance": 0,
"SecondsBehindMaster": 0,
"StepAll": 3,
"StepInfo": [
{
"Errors": [],
"StartTime": "2022-07-11 17:21:59",
"Status": "finished",
"StepId": "dumper",
"StepName": "源库导出",
"StepNo": 1,
"Percent": 100,
"StepMessage": "",
"Warnings": []
},
{
"Errors": [],
"StartTime": "2022-07-11 17:22:43",
"Status": "finished",
"StepId": "loader",
"StepName": "数据导入",
"StepNo": 2,
"Percent": 100,
"StepMessage": "",
"Warnings": []
},
{
"Errors": [],
"StartTime": "2022-07-11 17:23:26",
"Status": "running",
"StepId": "sinker",
```

```
"StepName": "同步增量",
"Percent": 0,
"StepMessage": "",
"StepNo": 3,
"Warnings": []
}
],
"StepNow": 3
},
"DstInfo": {
"AccessType": "cdb",
"DatabaseType": "mysql",
"ExtraAttr": [],
"Info": [
{
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnGwId": "",
"CvmInstanceId": "",
"DbKernel": "",
"EngineVersion": "",
"Host": "",
"InstanceId": "cdb-o7uph0cj",
"Password": "Rogerp@ssw0rd",
"Port": 0,
"Role": "",
"SubnetId": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
}
],
"NodeType": "simple",
"Region": "ap-guangzhou",
"Supplier": ""
},
"EndTime": "0000-00-00 00:00:00",
"ExpectRunTime": "",
"JobId": "dts-amm1jw5q",
"JobName": "test_config_api",
"RunMode": "immediate",
"SrcInfo": {
"AccessType": "intranet",
"DatabaseType": "mysql",
"ExtraAttr": [],
"Info": [
{
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnGwId": "",
"CvmInstanceId": "",
```

```
"DbKernel": "",
"EngineVersion": "5.7",
"Host": "9.123.210.37",
"InstanceId": "",
"Password": "Rogerp@ssw0rd",
"Port": 31035,
"Role": "",
"SubnetId": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
}
],
"NodeType": "simple",
"Region": "ap-guangzhou",
"Supplier": ""
},
"StartTime": "2022-07-11 17:20:56",
"Status": "readyComplete",
"Tags": [],
"TradeInfo": {
"BillingType": "billing",
"DealName": "2022071171500104792948 1",
"ExpireTime": "0000-00-00 00:00:00",
"OfflineTime": "0000-00-00 00:00:00",
"IsolateTime": "0000-00-00 00:00:00",
"InstanceClass": "xlarge",
"IsolateReason": "",
"LastDealName": "",
"OfflineReason": "",
"PayType": "postpay",
"TradeStatus": "normal"
},
"UpdateTime": "2022-07-11 17:26:40"
}
],
"RequestId": "asdfadfasdfasdf",
"TotalCount": 1
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee

- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.ControllerNotFoundError	异常错误导致当前接口未注册。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

购买迁移任务

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

购买迁移任务。购买成功后会返回随机生成的迁移任务id列表，也可以通过查询迁移任务任务列表接口 `DescribeMigrationJobs` 看到购买成功的实例Id。注意，一旦购买成功后源及目标数据库类型，源及目标实例地域不可修改。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateMigrationService。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SrcDatabaseType	是	String	源实例数据库类型，如mysql,redis,percona,mongodb,postgresql,sqlserver,mariadb,cynosdbmysql 示例值：mysql
DstDatabaseType	是	String	目标实例数据库类型，如mysql,redis,percona,mongodb,postgresql,sqlserver,mariadb,cynosdbmysql 示例值：mysql
SrcRegion	是	String	源实例地域，如：ap-guangzhou 示例值：ap-guangzhou
DstRegion	是	String	目标实例地域，如：ap-guangzhou。注意，目标地域必须和API请求地域保持一致。 示例值：ap-guangzhou
InstanceClass	是	String	实例规格，包括：small、medium、large、xlarge、2xlarge 示例值：large
Count	否	Integer	购买数量，范围为[1,15]，默认为1 示例值：1
JobName	否	String	迁移服务名称，最大长度128 示例值：migrate_test
Tags.N	否	Array of TagItem	标签信息

3. 输出参数

参数名称	类型	描述
JobIds	Array of String	下单成功随机生成的迁移任务id列表，形如：dts-c1f6rs21 注意：此字段可能返回 null，表示取不到有效值。 示例值：["dts-amm1jw5q"]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 购买迁移任务

购买迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateMigrationService
<公共请求参数>

{
"SrcDatabaseType": "mysql",
"DstDatabaseType": "mysql",
"InstanceClass": "xlarge",
"SrcRegion": "ap-guangzhou",
"DstRegion": "ap-guangzhou",
"Tags": [
{
"TagKey": "testkey",
"TagValue": "testvalue"
}
]
}
```

输出示例

```
{
"Response": {
"JobIds": [
"dts-1ewjspuw"
],
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

版权所有：腾讯云计算（北京）有限责任公司

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

配置迁移服务

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

配置迁移服务，配置成功后可通过 `CreateMigrationCheckJob` 创建迁移校验任务接口发起校验任务，只有校验通过才能启动迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrationJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id 示例值：dts-1ewjspuw
RunMode	是	String	运行模式，取值如：immediate(表示立即运行)、timed(表示定时运行) 示例值：immediate
MigrateOption	是	MigrateOption	迁移任务配置选项，描述任务如何执行迁移等一系列配置信息；字段下的RateLimitOption不可配置、如果需要修改任务的限速信息、请在任务运行后通过ModifyMigrateRateLimit接口修改
SrcInfo	是	DBEndpointInfo	源实例信息
DstInfo	是	DBEndpointInfo	目标实例信息
JobName	否	String	迁移任务名称，最大长度128 示例值：migrate_test
ExpectRunTime	否	String	期待启动时间，当RunMode取值为timed时，此值必填，形如："2006-01-02 15:04:05" 示例值：2022-07-11 17:20:56
Tags.N	否	Array of TagItem	标签信息
AutoRetryTimeRangeMinutes	否	Integer	自动重试的时间段、可设置5至720分钟、0表示不重试 示例值：5

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrationJob
<公共请求参数>

{
"JobId": "dts-efyc7e71",
"JobName": "dts-efyc7e71",
"RunMode": "immediate",
"ExpectRunTime": "2024-12-20 19:01:07",
"MigrateOption": {
"MigrateType": "fullAndIncrement",
"Consistency": {
"Mode": "full"
},
"DatabaseTable": {
"ObjectMode": "partial",
"Databases": [
{
"DbName": "db_1",
"NewDbName": "",
"SchemaName": "",
"NewSchemaName": "",
"DBMode": "partial",
"SchemaMode": "",
"TableMode": "partial",
"Tables": [
{
"TableName": "tb_1",
"NewTableName": "",
"TmpTables": [],
"TableEditMode": ""
}
],
"ViewMode": "partial",
"Views": [
{
"ViewName": "view_tb_1",
"NewViewName": ""
}
],
"RoleMode": "",
"Roles": [],
"FunctionMode": "",
"TriggerMode": "",
"EventMode": "",
"ProcedureMode": "",
"Functions": [],
"Procedures": [],
"Events": [],
"Triggers": []
}
],
"AdvancedObjects": []
},
```

```
"IsMigrateAccount": true,
"IsOverrideRoot": true,
"IsDstReadOnly": true,
"ExtraAttr": []
},
"SrcInfo": {
"Supplier": "",
"Region": "ap-qingyuan",
"AccessType": "cdb",
"DatabaseType": "mysql",
"NodeType": "",
"Info": [
{
"Role": "",
"DbKernel": "",
"Host": "",
"Port": 0,
"User": "tencent",
"Password": "tencent***",
"CvmInstanceId": "",
"UniqVpnGwId": "",
"UniqDcgId": "",
"InstanceId": "cdb-4gp2nzqy",
"CcnGwId": "",
"VpcId": "",
"SubnetId": "",
"EngineVersion": "8.0",
"Account": "",
"AccountRole": "",
"AccountMode": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"EncryptConn": "Encrypted"
}
],
"ExtraAttr": [],
"DatabaseNetEnv": ""
},
"DstInfo": {
"Supplier": "",
"Region": "ap-qingyuan",
"AccessType": "cdb",
"DatabaseType": "mysql",
"NodeType": "",
"Info": [
{
"Role": "",
"DbKernel": "",
"Host": "",
"Port": 0,
"User": "tencent",
"Password": "tencent***",
"CvmInstanceId": "",
"UniqVpnGwId": "",
"UniqDcgId": "",
"InstanceId": "cdb-4gp2nzqy",
```

```
"CcnGwId": "",
"VpcId": "",
"SubnetId": "",
"EngineVersion": "8.0",
"Account": "",
"AccountRole": "",
"AccountMode": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"EncryptConn": "Encrypted"
}
],
"ExtraAttr": [],
"DatabaseNetEnv": ""
},
"Tags": [],
"AutoRetryTimeRangeMinutes": 0
}
```

输出示例

```
{
"Response": {
"RequestId": "a3365b29-2042-4565-9206-8df9e2af1a17"
}
}
```

示例2 配置一个直连下云的redis迁移任务

1.源端为腾讯云redis实例，目标端为AWS实例并以公网的方式连接目标节点信息填写所有分片上的IP/域名、端口、密码等连接信息；注意一定是所有分片节点连接信息都写。
2.目标端以覆盖写的方式执行，如果要配置其他目标端写入模式，可以参考入参MigrateOption.ExtraAttr中的字段描述。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrationJob
<公共请求参数>

{
"JobId": "dts-n6uzsuxz",
"RunMode": "immediate",
"MigrateOption": {
"DatabaseTable": {
"ObjectMode": "all"
},
"MigrateType": "fullAndIncrement",
"ExtraAttr": [
{
"Key": "DstWriteMode",
"Value": "overwrite"
}
]
},
```

```
"SrcInfo": {
"Region": "ap-chengdu",
"AccessType": "cdb",
"DatabaseType": "redis",
"NodeType": "simple",
"Info": [
{
"Password": "admin12345",
"InstanceId": "crs-lktt4ls3"
}
]
},
"DstInfo": {
"Region": "ap-chengdu",
"AccessType": "extranet",
"DatabaseType": "redis",
"NodeType": "cluster-cache",
"Info": [
{
"Host": "44.223.107.125",
"Port": 6383
},
{
"Host": "44.223.107.125",
"Port": 6384
},
{
"Host": "44.223.107.125",
"Port": 6385
}
]
}
}
```

输出示例

```
{
"Response": {
"RequestId": "4d21a953-4a23-47f1-91a9-84df62f016f3"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee

- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
AuthFailure.AuthorizedOperationDenyError	认证拒绝错误。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InternalError.InternalHttpServerError	http请求访问出错。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.DataConvertError	数据转换错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

校验迁移任务

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

校验迁移任务，
在开始迁移前, 必须调用本接口创建校验迁移任务, 且校验成功后才能开始迁移. 校验的结果可以通过DescribeMigrationCheckJob查看，
校验成功后,迁移任务若有修改, 则必须重新校验并通过后, 才能开始迁移

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateMigrateCheckJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建校验迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateMigrateCheckJob
<公共请求参数>

{
"JobId": "dts-1kl0iy0v"
}
```

输出示例

```
{
"Response": {
```

```
"RequestId": "46b45da1-3e8e-4ef2-8de8-b1bffa027385"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
InternalError	内部错误。
InternalError.AddTaskError	添加异步任务失败。
InternalError.CeleryError	内部调度系统错误。
InternalError.CgwSystemError	cgw系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.LockError	锁冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。

错误码	描述
InvalidParameter.InstanceNotFound	实例不存在。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied	操作被拒绝。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.IntraNetUserNotTaggedError	自研上云用户创建迁移任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

查询迁移校验任务结果

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于创建校验后,获取校验的结果. 能查询到当前校验的状态和进度.
若通过校验, 则可调用'StartMigrateJob' 开始迁移.
若未通过校验, 则能查询到校验失败的原因. 请按照报错, 通过'ModifyMigrationJob'修改迁移配置或是调整源/目标实例的相关参数.

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrationCheckJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id 示例值：dts-1ewjspuw

3. 输出参数

参数名称	类型	描述
Status	String	校验任务执行状态，如：notStarted(未开始)、running(校验中)、failed(校验任务失败)、success(任务成功) 示例值：success
BriefMsg	String	校验任务结果输出简要信息 示例值：选择的库表不存在，请重新选择[存在输入参数table在源实例没有找到]
StepInfo	Array of CheckStep	检查步骤
CheckFlag	String	校验结果，如：checkPass(校验通过)、checkNotPass(校验未通过) 示例值：checkPass
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询迁移校验任务结果

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeMigrationCheckJob
<公共请求参数>
```

```
{
"JobId": "dts-1ewjspuw"
}
```


输出示例


```
{
"Response": {
"RequestId": "11d35c90-01bb-11ed-bad9-7b3bbe11abda",
"CheckFlag": "checkPass",
"BriefMsg": "success",
"Status": "success",
"StepInfo": [
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "ConnectDBCheck",
"StepMessage": "",
"StepName": "连接DB检查",
"StepNo": 0,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "VersionCheck",
"StepMessage": "",
"StepName": "版本检查",
"StepNo": 1,
"StepStatus": "pass"
},
{
"DetailCheckItems": [
{
"CheckItemName": "源实例权限检查",
"CheckResult": "warning",
"Description": "",
"ErrorLog": [
"您的授权情况：GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE ON *.* TO 'root'@'%' WITH GRANT OPTION。 源端账号权限范围大于要求的账号权限范围，腾讯云DTS不会使用超过必要的权限，请按照最小化原则给与授权：https://cloud.tencent.com/document/product/571/58700"
],
"FailureReason": "",
"HelpDoc": [
""
],
"SkipInfo": "",
"Solution": ""
}
],
"HasSkipped": false,
"StepId": "SrcPrivilegeCheck",
"StepMessage": "",
"StepName": "源实例权限检查",
```

```
"StepNo": 2,
"StepStatus": "warning"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "SimpleParamCheck",
"StepMessage": "",
"StepName": "部分实例参数检查",
"StepNo": 3,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "DstPrivilegeCheck",
"StepMessage": "",
"StepName": "目标实例权限检查",
"StepNo": 4,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "DstEmptyCheck",
"StepMessage": "",
"StepName": "目标实例内容冲突检查",
"StepNo": 5,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "DstSpaceCheck",
"StepMessage": "",
"StepName": "目标实例空间检查",
"StepNo": 6,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "BinlogParamCheck",
"StepMessage": "",
"StepName": "binlog参数检查",
"StepNo": 7,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "ConstraintCheck",
"StepMessage": "",
"StepName": "外键依赖检查",
"StepNo": 8,
"StepStatus": "pass"
},
```

```
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "ConstraintRefCheck",
"StepMessage": "",
"StepName": "外键部分库表依赖检查",
"StepNo": 9,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "ViewCheck",
"StepMessage": "",
"StepName": "视图检查",
"StepNo": 10,
"StepStatus": "pass"
},
{
"DetailCheckItems": [],
"HasSkipped": false,
"StepId": "WarningParamCheck",
"StepMessage": "",
"StepName": "警告项检查",
"StepNo": 11,
"StepStatus": "pass"
},
{
"DetailCheckItems": [
{
"CheckItemName": "周边检查",
"CheckResult": "warning",
"Description": "",
"ErrorLog": [
"skipped"
],
"FailureReason": "",
"HelpDoc": [
""
],
"SkipInfo": "",
"Solution": ""
}
],
"HasSkipped": true,
"StepId": "OptimizeCheck",
"StepMessage": "",
"StepName": "周边检查",
"StepNo": 12,
"StepStatus": "failed"
}
]
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

跳过迁移校验检查项

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于校验检查项不通过后，可进行跳过此校验项操作，后端将不再校验该项。任何校验步骤都是不应该跳过的，通过校验是能正确执行的前置条件。支持跳过的产品及链路的校验项可 参考文档。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：SkipCheckItem。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v
StepIds.N	是	Array of String	需要跳过校验项的步骤id，需要通过DescribeMigrationCheckJob接口返回StepInfo[i].StepId字段获取，例如：["OptimizeCheck"] 示例值：["OptimizeCheck"]
ForeignKeyFlag	否	String	当出现外键依赖检查导致校验不通过时、可以通过该字段选择是否迁移外键依赖，当StepIds包含ConstraintCheck且该字段值为shield时表示不迁移外键依赖、当StepIds包含ConstraintCheck且值为migrate时表示迁移外键依赖 示例值：shield

3. 输出参数

参数名称	类型	描述
Message	String	跳过的提示信息 示例值：skip item successfully
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 跳过迁移校验检查项

当校验不通过时、用户需要跳过校验失败项时、可通过这个接口完成跳过；也可通过这个接口设置是否迁移外键依赖。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: SkipCheckItem
<公共请求参数>
```

```
{
"JobId": "dts-1ewjspuw",
"StepIds": [
"OptimizeCheck"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "46b45da1-3e8e-4ef2-8de8-b1bffa027385",
"Message": "skip item successfully"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。

启动数据迁移任务

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（StartMigrationJob）用于启动迁移任务。调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 启动数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StartMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "bab78b9b-ce8a-4645-bfb2-5b03397d6ea0"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StartJobFailed	任务开始失败。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.AddTaskError	添加异步任务失败。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.LockError	锁冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。

错误码	描述
LimitExceeded.LimitExceededError	数量超出限制。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

完成数据迁移任务

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（CompleteMigrateJob）用于完成数据迁移任务。
选择采用增量迁移方式的任务，需要在迁移进度进入准备完成阶段后，调用本接口，停止迁移增量数据。
通过DescribeMigrationJobs接口查询到任务的状态为准备完成（Status="readyComplete"）时，此时可以调用本接口完成迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CompleteMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v
CompleteMode	否	String	完成任务的方式,仅支持旧版MySQL迁移任务。waitForSync-等待主从差距为0才停止,immediately-立即完成，不会等待主从差距一致。默认为waitForSync 示例值：waitForSync

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 完成数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CompleteMigrateJob
<公共请求参数>

{
"JobId": "dts-1kl0iy0v"
}
```

输出示例

```
{
"Response": {
"RequestId": "5268ea15-6636-44d1-89f9-e989d1c82e7d"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.CeleryError	内部调度系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

恢复暂停中的迁移任务

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

恢复一个暂停中的迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ContinueMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-xxxxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 恢复暂停中的迁移任务

任务处于暂停中时、可通过该接口继续任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ContinueMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "bab78b9b-ce8a-4645-bfb2-5b03397d6ea0"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。

查询可迁移的实例列表

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于查询支持迁移的云数据库实例

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrateDBInstances。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
DatabaseType	是	String	数据库类型，如mysql,redis等 示例值：mysql
MigrateRole	否	String	实例作为迁移的源还是目标,src(表示源)，dst(表示目标) 示例值：dst
InstanceId	否	String	云数据库实例ID 示例值：cdb-ktexpqjo
InstanceName	否	String	云数据库名称 示例值：cdb-name1
Limit	否	Integer	返回数量限制 示例值：20
Offset	否	Integer	偏移量 示例值：0
AccountMode	否	String	资源所属账号 为空值或self(表示本账号内资源)、other(表示其他账户资源) 示例值：self
TmpSecretId	否	String	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：AKIDXXXXXX
TmpSecretKey	否	String	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：vobJXXXXXX
TmpToken	否	String	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：MvIVXXXXXX

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	符合筛选条件的数量 示例值：1
Instances	Array of MigrateDBItem	实例列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询迁移实例列表

查询迁移实例列表

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeMigrateDBInstances
<公共请求参数>

{
"InstanceId": "cdb-eur39922",
"DatabaseType": "mysql",
"MigrateRole": "src"
}
```

输出示例

```
{
"Response": {
"TotalCount": 1,
"RequestId": "cafebabe-848392-di98382-dfu9832",
"Instances": []
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee

版权所有：腾讯云计算（北京）有限责任公司

- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter	缺少参数错误。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.PermissionDenied	操作未被授权。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter	未知参数错误。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

查询某个迁移任务详情

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询某个迁移任务详情

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrationDetail。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-amm1jw5q

3. 输出参数

参数名称	类型	描述
JobId	String	数据迁移任务ID 示例值：dts-amm1jw5q
JobName	String	数据迁移任务名称 示例值：dts-amm1jw5q
CreateTime	String	任务创建(提交)时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
UpdateTime	String	任务更新时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
StartTime	String	任务开始执行时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
EndTime	String	任务执行结束时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
BriefMsg	String	迁移任务简要错误信息 示例值：dial tcp : i/o timeout
Status	String	任务状态，取值为：created(创建完成)、checking(校验中)、checkPass(校验通过)、checkNotPass(校验不通过)、readyRun(准备运行)、running(任务运行中)、readyComplete(准备完成)、success(任务成功)、failed(任务失败)、stopping(中止中)、completing(完成中)、pausing(暂停中)、 manualPaused(已暂停) 示例值：running
Action	MigrateAction	任务操作信息
StepInfo	MigrateDetailInfo	迁移执行过程信息，在校验阶段显示校验过程步骤信息，在迁移阶段会显示迁移步骤信息

参数名称	类型	描述
SrcInfo	DBEndpointInfo	源实例信息
DstInfo	DBEndpointInfo	目标端信息
CompareTask	CompareTaskInfo	数据一致性校验结果
Tags	Array of TagItem	标签信息
RunMode	String	运行模式，取值如：immediate(表示立即运行)、timed(表示定时运行) 示例值：immediate
ExpectRunTime	String	期待启动时间，当RunMode取值为timed时，此值必填，形如："2006-01-02 15:04:05" 示例值：2022-07-11 16:20:49
MigrateOption	MigrateOption	迁移选项，描述任务如何执行迁移等一系列配置信息
CheckStepInfo	CheckStepInfo	校验任务运行详情
TradeInfo	TradeInfo	描述计费相关的信息
ErrorInfo	Array of ErrorInfoItem	任务错误信息
DumperResumeCtrl	String	全量导出可重入标识：enum::"yes"/"no"。yes表示当前任务可重入、no表示当前任务处于全量导出且不可重入阶段；如果在该值为no时重启任务导出流程不支持断点续传 示例值：yes
RateLimitOption	RateLimitOption	任务的限速信息
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询某个迁移任务详情

查询某个迁移任务详情

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeMigrationDetail
<公共请求参数>

{
"JobId": "dts-amm1jw5q"
}
```

输出示例

```
{
"Response": {
"Action": {
"AllAction": [
"view",
"modify",
"check",
"start",
"configure",
"stop",
```

```
"complete",
"resume",
"createCmpTask",
"isolate",
"offline",
"resize",
"recover"
],
"AllowedAction": [
"view",
"stop",
"complete",
"createCmpTask",
"view",
"isolate",
"resize"
]
},
"CheckStepInfo": {
"EndAt": "2022-07-11 17:19:25",
"Progress": {
"Message": "success",
"Percent": 100,
"Status": "success",
"StepAll": 12,
"Steps": [
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConnectDBCheck",
"StepMessage": "",
"StepName": "连接DB检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "VersionCheck",
"StepMessage": "",
"StepName": "版本检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SrcPrivilegeCheck",
"StepMessage": "",
"StepName": "源实例权限检查",
"StepNo": 0,
```

```
"Warnings": [
{
"HelpDoc": "",
"Message": "您的授权情况：GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE ON *.* TO 'root'@'%' WITH GRANT OPTION。 源端账号权限范围大于要求的账号权限范围，腾讯云DTS不会使用超过必要的权限，请按照最小化原则给与授权：https://cloud.tencent.com/document/product/571/58700",
"Solution": ""
}
]
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SimpleParamCheck",
"StepMessage": "",
"StepName": "部分实例参数检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstPrivilegeCheck",
"StepMessage": "",
"StepName": "目标实例权限检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstEmptyCheck",
"StepMessage": "",
"StepName": "目标实例内容冲突检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstSpaceCheck",
"StepMessage": "",
"StepName": "目标实例空间检查",
"StepNo": 0,
"Warnings": []
},
{
```

```
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "BinlogParamCheck",
"StepMessage": "",
"StepName": "binlog参数检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintCheck",
"StepMessage": "",
"StepName": "外键依赖检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintRefCheck",
"StepMessage": "",
"StepName": "外键部分库表依赖检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ViewCheck",
"StepMessage": "",
"StepName": "视图检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"StartTime": "",
"Status": "finished",
"StepId": "WarningParamCheck",
"StepMessage": "",
"StepName": "警告项检查",
"StepNo": 0,
"Warnings": []
},
{
"Errors": [],
"Percent": 0,
```

```
"StartTime": "",
"Status": "skipped",
"StepId": "OptimizeCheck",
"StepMessage": "",
"StepName": "周边检查",
"StepNo": 0,
"Warnings": [
{
"HelpDoc": "",
"Message": "skipped",
"Solution": ""
}
]
}
],
"StepNow": 12
},
"StartAt": "2022-07-11 17:18:57"
},
"CompareTask": {
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk",
"Status": "consistent"
},
"CreateTime": "2022-07-11 16:20:49",
"DstInfo": {
"AccessType": "cdb",
"DatabaseType": "mysql",
"ExtraAttr": [],
"Info": [
{
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnGwId": "",
"CvmInstanceId": "",
"DbKernel": "",
"EngineVersion": "",
"Host": "",
"InstanceId": "cdb-o7uph0cj",
"Password": "Abc_123456789",
"Port": 0,
"Role": "",
"SubnetId": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
}
],
"NodeType": "simple",
"Region": "ap-guangzhou",
"Supplier": ""
},
"EndTime": "0000-00-00 00:00:00",
```

```
"ErrorInfo": [],
"ExpectRunTime": "2022-07-11 17:16:50",
"JobId": "dts-amm1jw5q",
"JobName": "test_config_api",
"BriefMsg": "",
"MigrateOption": {
"Consistency": {
"Mode": "full"
},
"DatabaseTable": {
"Databases": [
{
"DBMode": "partial",
"DbName": "big100",
"EventMode": "",
"FunctionMode": "",
"NewDbName": "",
"NewSchemaName": "",
"ProcedureMode": "",
"RoleMode": "",
"Roles": [],
"SchemaMode": "",
"SchemaName": "",
"TableMode": "partial",
"Tables": [
{
"NewTableName": "",
"TableEditMode": "",
"TableName": "sbtest1",
"TmpTables": []
},
{
"NewTableName": "new_sbtest10",
"TableEditMode": "rename",
"TableName": "sbtest10",
"TmpTables": []
},
{
"NewTableName": "",
"TableEditMode": "",
"TableName": "sbtest100",
"TmpTables": []
}
],
"TriggerMode": "",
"ViewMode": "none",
"Views": []
},
{
"DBMode": "all",
"DbName": "db1",
"EventMode": "",
"FunctionMode": "",
"NewDbName": "",
"NewSchemaName": "",
"ProcedureMode": "",
"RoleMode": "",
```

```
"Roles": [],
"SchemaMode": "",
"SchemaName": "",
"TableMode": "",
"Tables": [],
"TriggerMode": "",
"ViewMode": "",
"Views": []
}
],
"AdvancedObjects": [
"function"
],
"ObjectMode": "partial"
},
"ExtraAttr": [],
"IsMigrateAccount": false,
"IsDstReadOnly": false,
"IsOverrideRoot": false,
"MigrateType": "fullAndIncrement"
},
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2",
"RunMode": "immediate",
"SrcInfo": {
"AccessType": "intranet",
"DatabaseType": "mysql",
"ExtraAttr": [],
"Info": [
{
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnGwId": "",
"CvmInstanceId": "",
"DbKernel": "",
"EngineVersion": "5.7",
"Host": "9.123.456.789",
"InstanceId": "",
"Password": "Abc_123456789",
"Port": 31035,
"Role": "",
"SubnetId": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
}
],
"NodeType": "simple",
"Region": "ap-guangzhou",
"Supplier": ""
},
"StartTime": "2022-07-11 17:20:56",
"Status": "readyComplete",
```

```
"StepInfo": {
"MasterSlaveDistance": 0,
"SecondsBehindMaster": 0,
"StepAll": 3,
"StepInfo": [
{
"StepMessage": "",
"Percent": 100,
"Errors": [],
"StartTime": "2022-07-11 17:21:59",
"Status": "finished",
"StepId": "dumper",
"StepName": "源库导出",
"StepNo": 1,
"Warnings": []
},
{
"StepMessage": "",
"Percent": 100,
"Errors": [],
"StartTime": "2022-07-11 17:22:43",
"Status": "finished",
"StepId": "loader",
"StepName": "数据导入",
"StepNo": 2,
"Warnings": []
},
{
"StepMessage": "",
"Percent": 100,
"Errors": [],
"StartTime": "2022-07-11 17:23:26",
"Status": "running",
"StepId": "sinker",
"StepName": "同步增量",
"StepNo": 3,
"Warnings": []
}
],
"StepNow": 3
},
"Tags": [],
"TradeInfo": {
"BillingType": "billing",
"DealName": "2022071171500104792948 1",
"ExpireTime": "0000-00-00 00:00:00",
"InstanceClass": "xlarge",
"IsolateReason": "",
"LastDealName": "",
"OfflineReason": "",
"PayType": "postpay",
"TradeStatus": "normal",
"OfflineTime": "",
"IsolateTime": ""
},
"UpdateTime": "2022-07-11 19:53:03",
"DumperResumeCtrl": "no"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.ControllerNotFoundError	异常错误导致当前接口未注册。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。

错误码	描述
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

下线数据迁移任务

最近更新时间：2025-04-25 01:29:17

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

下线数据迁移任务。计费任务必须先调用隔离(IsolateMigrateJob)接口，且只有是**已隔离**状态下，才能调用此接口销毁任务。对于不计费任务，调用隔离(IsolateMigrateJob)接口删除任务操作。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DestroyMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	任务id 示例值：dts-j7bt5sid

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 下线数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DestroyMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

隔离数据迁移任务

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

隔离退还数据迁移服务。调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。对于计费任务，在任务隔离后可进行解除隔离(RecoverMigrationJob)操作或直接进行下线销毁(DestroyMigrateJob)操作。对于不计费任务，调用此接口会直接销毁任务，无法进行恢复操作。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：IsolateMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id 示例值：dts-amm1jw5q

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 隔离(销毁)数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: IsolateMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

调整实例规格

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

调整实例规格，此接口只支持按量计费任务的调整。调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrateJobSpec。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id 示例值：dts-j7bt5sid
NewInstanceClass	是	String	新实例规格大小，包括：micro、small、medium、large、xlarge、2xlarge 示例值：xlarge

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 调整实例规格

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrateJobSpec
<公共请求参数>

{
"JobId": "dts-j7bt5sid",
"NewInstanceClass": "xlarge"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

修改迁移名称

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改迁移任务名

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrateName。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务id 示例值：dts-7itt1j4w
JobName	是	String	修改后的迁移任务名 示例值：userDefine

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改迁移名称

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrateName
<公共请求参数>

{
"JobName": "userDefine",
"JobId": "dts-7itt1j4w"
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改任务运行时属性

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改任务运行时属性，此接口不同于配置类接口，不会进行状态机判断。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrateRuntimeAttribute。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务id，如：dts-2rgv0f09 示例值：dts-7itt1j4w

OtherOptions.N	是	Array of KeyValuePairOption	需要修改的属性，此结构设计为通用结构，用于屏蔽多个业务的定制属性。
			例如对于Redis:
			{
			"Key": "DstWriteMode", //目标库写入模式
			"Value": "normal" //clearData(清空目标实例数据)、overwrite(以覆盖写的方式执行任务)、normal(跟正常流程一样，不做额外动作，默认为此值)
			},
			{
			"Key": "IsDstReadOnly", //是否在迁移时设置目标库只读
			"Value": "true" //true(设置只读)、false(不设置只读)
			}

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改Redis目标库写入模式

修改Redis目标库写入模式，填写对应参数即可，也可同时修改多个参数。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrateRuntimeAttribute
<公共请求参数>
```

```
{
"JobId": "dts-2rgv0f09",
"OtherOptions": [
{
"Key": "DstWriteMode",
"Value": "clearData"
}
]
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.ExecHttpRequestError	http请求出错。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
OperationDenied.OperationDeniedError	操作被拒绝。

错误码	描述
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnknownParameter.UnknownParameterError	未知参数。

暂停迁移任务

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

暂停一个迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：PauseMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-xxxxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 暂停迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: PauseMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "bab78b9b-ce8a-4645-bfb2-5b03397d6ea0"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。

解除隔离数据迁移任务

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

解除隔离数据迁移任务，用户手动发起隔离后的手动解隔离，只有任务状态为已隔离(手动操作)状态下才能触发此操作。调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：RecoverMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id 示例值：dts-7o16y2qy

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 解除隔离数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: RecoverMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalHttpServerError	http请求访问出错。
ResourceNotFound.ResourceNotFoundError	资源未找到。

重试迁移任务

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

重试数据迁移任务，针对异常情况可进行重试，对于redis在失败时也可重试。注意：此操作跳过校验阶段，直接重新发起任务，相当于从StartMigrationJob开始执行。调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResumeMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v
ResumeOption	否	String	恢复任务的模式，目前的取值有：clearData 清空目标实例数据，overwrite 以覆盖写的方式执行任务，normal 跟正常流程一样，不做额外动作；注意，clearData、overwrite仅对redis生效，normal仅针对非redis链路生效 示例值：overwrite

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 重试失败的数据迁移任务

在任务失败时，以覆盖写等模式重试已经失败的数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ResumeMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid",
"ResumeOption": "normal"
}
```

输出示例

```
{
"Response": {
"RequestId": "22f6b9ca-e94d-4a4d-b48f-a5133e791374"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValue	参数值错误。
LimitExceeded.LimitExceededError	数量超出限制。
OperationDenied.OperationDeniedError	操作被拒绝。

错误码	描述
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

终止数据迁移任务

最近更新时间：2025-04-25 01:29:15

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（StopMigrateJob）用于终止数据迁移任务。
调用此接口后可通过查询迁移服务列表接口 `DescribeMigrationJobs` 来查询当前任务状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StopMigrateJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 停止数据迁移任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StopMigrateJob
<公共请求参数>

{
"JobId": "dts-j7bt5sid"
}
```

输出示例

```
{
"Response": {
"RequestId": "22f6b9ca-e94d-4a4d-b48f-a5133e791374"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。

错误码	描述
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改迁移任务的传输速率

最近更新时间：2025-04-25 01:29:16

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

用户在发现迁移任务对用户的数据库的负载影响较大时、可通过该接口限制任务的传输速率

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrateRateLimit。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务ID 示例值：dts-saasxa
DumpThread	否	Integer	迁移任务全量导出线程数、有效值为 1-16 示例值：8
DumpRps	否	Integer	迁移全量导出的 Rps 限制、需要大于 0 示例值：400000
LoadThread	否	Integer	迁移任务全量导入线程数、有效值为 1-16 示例值：8
SinkerThread	否	Integer	迁移任务增量导入线程数、有效值为 1-128 示例值：32
LoadRps	否	Integer	全量导入Rps限制 示例值：400000

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改全量导出线程数

修改全量导出线程数成功

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyMigrateRateLimit
```

```
<公共请求参数>

{
"JobId": "dts-lfjkazze",
"DumpThread": 12
}
```

输出示例

```
{
"Response": {
"RequestId": "ce7594b5-3fbc-40d8-a514-c35186cef95a"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 公共错误码。

数据同步相关接口
查询同步任务信息

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询同步任务信息

默认接口请求频率限制：100次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSyncJobs。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	同步任务id，如sync-werwfs23 示例值：sync-n3gh7md9
JobIds.N	否	Array of String	同步任务id列表，如sync-werwfs23 示例值：["sync-n3gh7md9"]
JobName	否	String	同步任务名 示例值：sync-n3gh7md9
Order	否	String	排序字段，可以取值为CreateTime 示例值：CreateTime
OrderSeq	否	String	排序方式，升序为ASC，降序为DESC，默认为CreateTime降序 示例值：DESC
Offset	否	Integer	偏移量，默认为0 示例值：0
Limit	否	Integer	返回同步任务实例数量，默认20，有效区间[1,100] 示例值：2
Status.N	否	Array of String	状态集合，如Initialized,CheckPass,Running,ResumableErr,Stopped 示例值：Running
RunMode	否	String	运行模式，如Immediate:立即运行，Timed:定时运行 示例值：Immediate
JobType	否	String	任务类型，如mysql2mysql：msyql同步到mysql 示例值：mysql2mysql
PayMode	否	String	付费类型，PrePay：预付费，PostPay：后付费 示例值：PostPay
TagFilters.N	否	Array of TagFilter	tag
SrcInfoPattern	否	String	源端数据库连接信息，可以输入实例ID或者IP等 示例值：cdb-jk09iy6a

参数名称	必选	类型	描述
DstInfoPattern	否	String	目标端数据库连接信息，可以输入实例ID或者IP等 示例值：cdb-jk09iy6a

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	任务数目 示例值：2
JobList	Array of SyncJobInfo	任务详情数组
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询同步任务列表

查询同步任务列表

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeSyncJobs
<公共请求参数>

{
"JobId": "sync-bkm3mppi"
}
```

输出示例

```
{
"Response": {
"JobList": [
{
"Actions": [
"pause",
"stop",
"limitRate",
"view",
"resize",
"isolate",
"postPay2prePay",
"configureOptObj"
],
"AllActions": [
"view",
"configure",
"check",
"start",
"stop",
"resume",
```

```
"pause",
"continue",
"limitRate",
"resize",
"isolate",
"offline",
"recover",
"postPay2prePay",
"configureOptObj",
"checkOptObj",
"startOptObj",
"createCmpTask"
],
"AutoRenew": 0,
"AutoRetryTimeRangeMinutes": 0,
"CreateTime": "2023-12-18 20:55:27",
"Detail": {
"CauseOfCompareDisable": "当前任务开启了DML过滤()。",
"CurrentStepProgress": 0,
"ErrInfo": {
"Message": "",
"Reason": "",
"Solution": ""
},
"MasterSlaveDistance": 0,
"Message": "",
"Progress": -1,
"SecondsBehindMaster": 0,
"StepAll": 4,
"StepInfos": [
{
"Errors": null,
"Progress": -1,
"StartTime": "2023-12-20 17:03:39",
"Status": "finished",
"StepId": "schema-init",
"StepName": "增量库表初始化",
"StepNo": 1,
"Warnings": []
},
{
"Errors": null,
"Progress": -1,
"StartTime": "2023-12-20 17:03:49",
"Status": "finished",
"StepId": "dumper",
"StepName": "全量导出",
"StepNo": 2,
"Warnings": []
},
{
"Errors": null,
"Progress": -1,
"StartTime": "2023-12-20 17:07:50",
"Status": "finished",
"StepId": "loader",
"StepName": "全量导入",
```

```
"StepNo": 3,
"Warnings": []
},
{
"Errors": null,
"Progress": -1,
"StartTime": "2023-12-20 17:16:49",
"Status": "running",
"StepId": "sinker",
"StepName": "同步增量",
"StepNo": 4,
"Warnings": []
}
],
"StepNow": 4
},
"DstAccessType": "noProxy",
"DstDatabaseType": "mysql",
"DstInfo": {
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnId": "",
"CcnOwnerUin": "",
"CvmInstanceId": "",
"DatabaseNetEnv": "",
"DbKernel": "",
"DbName": "",
"EncryptConn": "",
"EngineVersion": "",
"InstanceId": "",
"Ip": "11.141.232.156",
"Password": "",
"Port": 3306,
"Region": "ap-guangzhou",
"Role": "",
"RoleExternalId": "",
"SubnetId": "",
"Supplier": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
},
"DstInfos": {
"AccessType": "",
"DatabaseType": "",
"Info": null,
"Region": ""
},
"DstNodeType": "single",
"DstRegion": "ap-qingyuan",
"DumperResumeCtrl": "yes",
"EndTime": "0000-00-00 00:00:00",
```

```
"ExpectRunTime": "0000-00-00 00:00:00",
"ExpireTime": "0000-00-00 00:00:00",
"InstanceClass": "medium",
"JobId": "sync-09vy1os1",
"JobName": "roger_test",
"Objects": {
"Databases": [
{
"DbMode": "All",
"DbName": "db_big",
"EventMode": "",
"FunctionMode": "",
"NewDbName": "",
"NewSchemaName": "",
"ProcedureMode": "",
"SchemaName": "",
"TableMode": "All",
"TriggerMode": "",
"ViewMode": "All"
}
],
"Mode": "Partial",
"OnlineDDL": {
"Status": ""
}
},
"OfflineTime": "0000-00-00 00:00:00",
"OptObjStatus": "Normal",
"Options": {
"AddAdditionalColumn": false,
"AutoRetryTimeRangeMinutes": 0,
"ConflictHandleOption": {
"ConditionColumn": "",
"ConditionOperator": "",
"ConditionOrderInSrcAndDst": ""
},
"ConflictHandleType": "Cover",
"DdlOptions": null,
"DealOfExistSameTable": "ExecuteAfterIgnore",
"FilterBeginCommit": false,
"FilterCheckpoint": false,
"InitType": "Full",
"KafkaOption": {
"DDLTopicName": "",
"DataType": "",
"TopicRules": [],
"TopicType": ""
},
"OpTypes": null,
"RateLimitOption": {
"CurrentDumpRps": 0,
"CurrentDumpThread": 8,
"CurrentLoadRps": 0,
"CurrentLoadThread": 8,
"CurrentSinkerThread": 32,
"DefaultDumpRps": 0,
"DefaultDumpThread": 8,
```

```
"DefaultLoadRps": 0,
"DefaultLoadThread": 8,
"DefaultSinkerThread": 32,
"HasUserSetRateLimit": "no"
},
"StartPosition": ""
},
"PayMode": "PostPay",
"RunMode": "Immediate",
"Specification": "",
"SrcAccessType": "noProxy",
"SrcDatabaseType": "mysql",
"SrcInfo": {
"Account": "",
"AccountMode": "",
"AccountRole": "",
"CcnId": "",
"CcnOwnerUin": "",
"CvmInstanceId": "",
"DatabaseNetEnv": "",
"DbKernel": "",
"DbName": "",
"EncryptConn": "",
"EngineVersion": "",
"InstanceId": "",
"Ip": "11.141.232.195",
"Password": "",
"Port": 3306,
"Region": "ap-guangzhou",
"Role": "",
"RoleExternalId": "",
"SubnetId": "",
"Supplier": "",
"TmpSecretId": "",
"TmpSecretKey": "",
"TmpToken": "",
"UniqDcgId": "",
"UniqVpnGwId": "",
"User": "root",
"VpcId": ""
},
"SrcInfos": {
"AccessType": "",
"DatabaseType": "",
"Info": null,
"Region": ""
},
"SrcNodeType": "single",
"SrcRegion": "ap-qingyuan",
"StartTime": "2023-12-19 15:54:18",
"Status": "Running",
"Tags": [],
"TradeStatus": "Normal"
}
],
"RequestId": "268b0711-a3b2-4692-8c91-88d743bc8b9f",
"TotalCount": 1
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.InternalErrorError	内部错误。
InternalError.ProtocolError	通信协议错误。
InvalidParameter	参数错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。

创建同步任务

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

创建一个同步任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
PayMode	是	String	付款类型, 如：PrePay(表示包年包月)、PostPay(表示按时按量) 示例值：PrePay
SrcDatabaseType	是	String	源端数据库类型,如mysql,mariadb,percona,postgresql,cynosdbmysql(表示TDSQL-C MySQL),tdpg(TDSQL PostgreSQL版),tdsqlmysql,tdstore(表示TDSQL TDStore版)等。 示例值：mysql
SrcRegion	是	String	源端数据库所在地域,如ap-guangzhou 示例值：ap-guangzhou
DstDatabaseType	是	String	目标端数据库类型,如mysql,mariadb,percona,cynosdbmysql(表示TDSQL-C MySQL),tdpg(TDSQL PostgreSQL版),tdsqlmysql,kafka,tdstore(表示TDSQL TDStore版)等。 示例值：mysql
DstRegion	是	String	目标端数据库所在地域,如ap-guangzhou 示例值：ap-guangzhou
Specification	否	String	同步任务规格，Standard:标准版 示例值：Standard
TimeSpan	否	Integer	购买时长（单位：月），当PayMode值为PrePay则此项配置有意义，默认为1月，取值范围为[1,100] 示例值：1
Tags.N	否	Array of TagItem	标签信息
Count	否	Integer	一次购买的同步任务数量，取值范围为[1, 10]，默认为1 示例值：1
AutoRenew	否	Integer	自动续费标识，当PayMode值为PrePay则此项配置有意义，取值为：1（表示自动续费）、0（不自动续费，默认为此值） 示例值：1
InstanceClass	否	String	同步链路规格，如micro,small,medium,large，默认为medium 示例值：large
JobName	否	String	同步任务名称 示例值：test_job
ExistedJobId	否	String	创建类似任务的现有任务Id

参数名称	必选	类型	描述
			示例值：sync-1dd3rjvt

3. 输出参数

参数名称	类型	描述
JobIds	Array of String	同步任务ids 示例值：sync-btso140
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建一个同步任务

创建一个同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateSyncJob
<公共请求参数>

{
"Count": 1,
"TimeSpan": "1",
"Specification": "Standard",
"PayMode": "PrePay",
"DstDatabaseType": "mysql",
"DstRegion": "ap-guangzhou",
"SrcDatabaseType": "mysql",
"SrcRegion": "ap-guangzhou",
"ExistedJobId": "sync-xascasas"
}
```

输出示例

```
{
"Response": {
"JobIds": [
"sync-eivc87yk"
],
"RequestId": "03a378be-372a-478b-beb7-19e4dd5027e2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.InternalTradeError	调用计费服务失败。
InternalError.NotEnoughMoneyError	用户余额不足。
InternalError.UnknownError	未知的内部错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.NotEnoughMoneyError	用户余额不足。
OperationDenied.OperationDeniedError	操作被拒绝。

版权所有：腾讯云计算（北京）有限责任公司

配置同步任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

配置一个同步任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ConfigureSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步实例id（即标识一个同步作业），形如sync-werwfs23 示例值：sync-werwfs23
SrcAccessType	是	String	源端接入类型，cdb(云数据库)、cvm(云主机自建)、vpc(私有网络)、extranet(外网)、vpncloud(vpn接入)、dcg(专线接入)、ccn(云联网)、intranet(自研上云),注意具体可选值依赖当前链路 示例值：cdb
DstAccessType	是	String	目标端接入类型，cdb(云数据库)、cvm(云主机自建)、vpc(私有网络)、extranet(外网)、vpncloud(vpn接入)、dcg(专线接入)、ccn(云联网)、intranet(自研上云)、ckafka(CKafka实例),注意具体可选值依赖当前链路 示例值：extranet
Objects	是	Objects	同步库表对象信息
JobName	否	String	同步任务名称 示例值：sync-werwfs23
JobMode	否	String	枚举值是 liteMode 和 fullMode ，分别对应精简模式或正常模式 示例值：fullMode
RunMode	否	String	运行模式，取值如：Immediate(表示立即运行，默认为此项值)、Timed(表示定时运行) 示例值：Immediate
ExpectRunTime	否	String	期待启动时间，当RunMode取值为Timed时，此值必填，形如："2006-01-02 15:04:05" 示例值：2022-12-06 00:00:00
SrcConnectType	否	String	源端tdsql连接方式：proxy-通过tdsql proxy主机访问各个set节点，注意只有在自研上云的网络环境下才能通过这种方式连接，SrcInfos中只需要提供proxy主机信息。set-直连set节点，如选择直连set方式，需要正确填写proxy主机信息及所有set节点信息。源端是tdsqlmysql类型必填。 示例值：proxy
SrcInfo	否	Endpoint	源端信息，单机版类型数据库配置使用，且SrcNodeType传single。例如mysql、percona、mariadb等。
SrcInfos	否	SyncDBEndpointInfos	源端信息，分布式类型数据库配置使用，且SrcNodeType传cluster。例如分布式数据库tdsqlmysql等。

参数名称	必选	类型	描述
SrcNodeType	否	String	枚举值：cluster、single。源库为单节点数据库使用single，多节点使用cluster 示例值：cluster
DstInfo	否	Endpoint	目标端信息，单机版类型数据库配置使用，且SrcNodeType传single。例如mysql、percona、mariadb等。
DstInfos	否	SyncDBEndpointInfos	目标端信息，分布式类型数据库配置使用，且SrcNodeType传cluster。例如分布式数据库tdsqlmysql等。
DstNodeType	否	String	枚举值：cluster、single。目标库为单节点数据库使用single，多节点使用cluster 示例值：single
Options	否	Options	同步任务选项；该字段下的RateLimitOption暂时无法生效、如果需要修改限速、可通过ModifySyncRateLimit接口完成限速
AutoRetryTimeRangeMinutes	否	Integer	自动重试的时间段、可设置5至720分钟、0表示不重试 示例值：5

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置同步任务

配置一个同步整库的同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ConfigureSyncJob
<公共请求参数>

{
"DstInfo": {
"Ip": "9.123.210.46",
"Password": "Test1234",
"User": "root",
"Port": 30654
},
"SrcInfo": {
"Ip": "9.123.211.23",
"Password": "Test1234",
"User": "root",
"Port": "30645"
},
"ExpectRunTime": "0000-00-00 00:00:01",
"DstAccessType": "noProxy",
"JobName": "truexu-api",
"RunMode": "Immediate",
"Objects": {
"Mode": "Partial",
"Databases": [
{
```

```
"DbName": "db1",
"DbMode": "All",
"TableMode": "All",
"Tables": null,
"ViewMode": "All",
"Views": null,
"FunctionMode": "All",
"TriggerMode": "",
"EventMode": "",
"ProcedureMode": "All",
"Functions": null,
"Procedures": null,
"Events": null,
"Triggers": null
}
]
},
"JobId": "sync-7r1cz016",
"Options": {
"DealOfExistSameTable": "ExecuteAfterIgnore",
"OpTypes": [
"Update"
],
"DdlOptions": [
{
"DdlObject": "Database",
"DdlValue": [
"Create",
"Alter"
]
}
],
"InitType": "Full",
"ConflictHandleOption": {
"ConditionColumn": "ts",
"ConditionOperator": ">",
"ConditionOrderInSrcAndDst": "<"
}
},
"SrcAccessType": "noProxy"
}
```


输出示例


```
{
"Response": {
"RequestId": "8d495a08-5f8f-411b-ae4d-cc6da2475b9c"
}
}
```


示例2 配置同步任务2

配置一个同步包含条件覆盖的任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ConfigureSyncJob
<公共请求参数>

{
"DstInfo": {
"Ip": "100.88.178.199",
"Password": "Test1234",
"User": "root",
"Port": 20152
},
"SrcInfo": {
"Ip": "100.88.166.23",
"Password": "Test1234",
"User": "root",
"Port": "20126"
},
"ExpectRunTime": "2021-12-08 11:59:05",
"DstAccessType": "noProxy",
"JobName": "truex",
"RunMode": "Immediate",
"Objects": {
"Mode": "Partial",
"Databases": [
{
"Tables": [
{
"FilterCondition": "id>1",
"TableName": "tb1",
"NewTableName": "tb2"
}
],
"Views": [
{
"ViewName": "vw1",
"NewViewName": "vw2"
}
],
"DbName": "db1",
"ViewMode": "Partial",
"TableMode": "Partial",
"DbMode": "Partial"
}
]
},
"JobId": "sync-31t047r6",
"Options": {
"DdlOptions": [
{
"DdlObject": "Database",
"DdlValue": [
"Drop"
]
}
```

```
],
"ConflictHandleType": "ReportError",
"DealOfExistSameTable": "ReportErrorAfterCheck",
"ConflictHandleOption": {
"ConditionColumn": "ts",
"ConditionOperator": ">",
"ConditionOrderInSrcAndDst": ">"
},
"InitType": "Full",
"OpTypes": [
"Insert",
"Update"
],
"AddAdditionalColumn": "false"
},
"SrcAccessType": "noProxy"
}
```

输出示例

```
{
"Response": {
"RequestId": "59dd9972-00e7-4ce3-8f33-72d13bc64400"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalHttpServerError	http请求访问出错。

错误码	描述
InternalError.UnknownError	未知的内部错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

校验同步任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

校验同步任务，检查必要参数和周边配置。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateCheckSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-n8deuq7e

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 校验同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCheckSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "d2b90690-3f7f-4650-93ab-62f5c4ddfe31"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.UnknownError	未知的内部错误。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnsupportedOperation	操作不支持。
UnsupportedOperation.IntraNetUserNotTaggedError	自研上云用户创建迁移任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。
UnsupportedOperation.IntraNetUserNotTaggedForSyncJobError	自研上云用户创建同步任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

查询同步校验任务结果

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询同步校验任务结果，检查必要参数和周边配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeCheckSyncJobResult。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	同步实例id（即标识一个同步作业），形如sync-werwfs23，此值必填 示例值：sync-werwfs23

3. 输出参数

参数名称	类型	描述
Status	String	校验任务执行状态，如：notStarted(未开始)、running(校验中)、failed(校验任务失败)、success(任务成功) 示例值：running
StepCount	Integer	步骤总数 示例值：14
StepCur	Integer	当前所在步骤 示例值：14
Progress	Integer	总体进度，范围为[0,100] 示例值：100
StepInfos	Array of StepInfo	步骤信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：StepInfos
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询同步任务校验结果

任务发起校验为异步操作，需要通过此接口查询同步任务校验结果

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
```

```
X-TC-Action: DescribeCheckSyncJobResult
<公共请求参数>

{
"JobId": "sync-bllmn8x9"
}
```

输出示例


```
{
"Response": {
"Progress": 100,
"RequestId": "948377d0-b8a2-11ed-b8ff-e9277ed2f336",
"Status": "success",
"StepCount": 14,
"StepCur": 14,
"StepInfos": [
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConnectDBCheck",
"StepName": "连接DB检查",
"StepNo": 1,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "OptimizeCheck",
"StepName": "周边检查",
"StepNo": 2,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "VersionCheck",
"StepName": "版本检查",
"StepNo": 3,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SrcPrivilegeCheck",
"StepName": "源实例权限检查",
"StepNo": 4,
"Warnings": [
```

```
{
"Code": "Warning: SrcPrivilegeCheck",
"HelpDoc": "",
"Message": "您的授权情况：GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE ON *.* TO 'root'@'%' WITH GRANT OPTION。 源端账号权限范围大于要求的账号权限范围，腾讯云DTS不会使用超过必要的权限，请按照最小化原则给与授权：https://cloud.tencent.com/document/product/571/58700",
"SkipInfo": "",
"Solution": ""
}
]
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SimpleParamCheck",
"StepName": "部分实例参数检查",
"StepNo": 5,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstPrivilegeCheck",
"StepName": "目标实例权限检查",
"StepNo": 6,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstEmptyCheck",
"StepName": "目标实例内容冲突检查",
"StepNo": 7,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstSpaceCheck",
"StepName": "目标实例空间检查",
"StepNo": 8,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
```

```
"Status": "finished",
"StepId": "BinlogParamCheck",
"StepName": "binlog参数检查",
"StepNo": 9,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintCheck",
"StepName": "外键依赖检查",
"StepNo": 10,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintRefCheck",
"StepName": "外键部分库表依赖检查",
"StepNo": 11,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ViewCheck",
"StepName": "视图检查",
"StepNo": 12,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "WarningParamCheck",
"StepName": "警告项检查",
"StepNo": 13,
"Warnings": [
{
"Code": "Warn: WarningParamCheck",
"HelpDoc": "",
"Message": "对于既没有主键、也没有不能为null的唯一键的表，有数据重复的风险。 请参考 https://cloud.tencent.com/document/product/571/58
739",
"SkipInfo": "",
"Solution": ""
},
{
"Code": "Warn: WarningParamCheck",
"HelpDoc": "",
"Message": "源实例中sql_mode为 'ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZE
```

```
RO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION'， 目标实例中sql_mode为 'ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DAT
E,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION'。建议设置为相同",
"SkipInfo": "",
"Solution": ""
}
]
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "AdvancedStoreObjectCheck",
"StepName": "高级对象检查",
"StepNo": 14,
"Warnings": []
}
]
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameterValue.InvalidParameterValueError	非法参数。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFoundError	资源未找到。

跳过同步校验检查项

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于校验检查项不通过后，可进行跳过此校验项操作，后端将不再校验该项。任何校验步骤都是不应该跳过的，通过校验是能正确执行的前置条件。支持跳过的产品及链路的校验项可 参考文档。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：SkipSyncCheckItem。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务id，如：sync-4ddgid2 示例值：sync-4ddgid2
StepIds.N	是	Array of String	需要跳过校验项的步骤id，需要通过`DescribeCheckSyncJobResult`接口返回StepInfos[i].StepId字段获取，例如：["OptimizeCheck"] 示例值：["OptimizeCheck"]

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 跳过同步校验检查项

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: SkipSyncCheckItem
<公共请求参数>

{
"JobId": "sync-4ddgid2",
"StepIds": [
"OptimizeCheck"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "46b45da1-3e8e-4ef2-8de8-b1bffa027385"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
InternalError	内部错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

启动同步任务

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

启动同步任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	同步任务id 示例值：sync-werwfs23

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 启动一个同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StartSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "31197c38-7764-49f6-845c-edbc5aa9294e"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.UnknownError	未知的内部错误。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。

隔离同步任务

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

隔离同步任务，隔离后可通过查询同步任务信息接口DescribeSyncJobs获取隔离后状态。在任务隔离后可进行解除隔离(RecoverSyncJob)操作或直接进行下线操作。对于不计费任务，调用此接口后会直接删除任务，无法进行恢复操作。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：IsolateSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-werwfs23

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 隔离一个同步任务

隔离一个正在计费的同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: IsolateSyncJob
<公共请求参数>

{
"JobId": "sync-kl5qyyf4"
}
```

输出示例

```
{
"Response": {
```

```
"RequestId": "df239e4f-7359-473b-a073-144ab9964350"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalTradeError	调用计费服务失败。
InternalError.UnknownError	未知的内部错误。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

解除隔离同步任务

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

解除隔离同步任务，任务在已隔离状态下可调用该接口解除隔离状态任务，同时可通过查询同步任务信息接口DescribeSyncJobs，获取操作后状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：RecoverSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步实例id（即标识一个同步作业），形如sync-werwfs23 示例值：sync-werwfs23

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 解除隔离同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: RecoverSyncJob
<公共请求参数>

{
"JobId": "sync-kl5qyyf4"
}
```

输出示例

```
{
"Response": {
"RequestId": "710040d7-3a04-44cb-a56e-0023cf0692ef"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValue	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。

错误码	描述
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

暂停同步任务

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

暂停处于同步中的数据同步任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：PauseSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-xxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 暂停一个同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: PauseSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "31197c38-7764-49f6-845c-edbc5aa9294e"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。

恢复被暂停的同步任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

恢复处于已暂停状态的数据同步任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ContinueSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-xxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 暂停一个同步任务

同步任务出于已暂停状态时、可通过这个接口恢复任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ContinueSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "31197c38-7764-49f6-845c-edbc5aa9294e"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

重试同步任务

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

重试同步任务，部分可恢复报错情况下，可通过该接口重试同步任务，可通过查询同步任务信息接口DescribeSyncJobs，获取操作后状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResumeSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-09vy1os1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 重试同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ResumeSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "91f8a1b0-8e41-4696-a24a-95e853da7071"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.UnknownError	未知的内部错误。

结束同步任务

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

结束同步任务，操作后可通过查询同步任务信息接口DescribeSyncJobs，获取操作后的状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StopSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-n8deuq7e

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 终止同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StopSyncJob
<公共请求参数>

{
"JobId": "sync-7r1cz016"
}
```

输出示例

```
{
"Response": {
"RequestId": "ead9fdc1-baa8-4a09-aaa1-c2012690c976"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.UnknownError	未知的内部错误。
ResourceNotFound.ResourceNotFoundError	资源未找到。

下线同步任务

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

下线同步任务，任务在已隔离状态下可以通过此操作进行任务下线，即彻底删除任务。下线操作后可通过查询同步任务信息接口DescribeSyncJobs获取任务列表查看状态，此操作成功后无法看到此任务表示下线成功。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DestroySyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-werwfs23

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 销毁一个同步任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DestroySyncJob
<公共请求参数>

{
"JobId": "sync-kl5qyyf4"
}
```

输出示例

```
{
"Response": {
"RequestId": "710040d7-3a04-44cb-a56e-0023cf0692ef"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalTradeError	调用计费服务失败。
InternalError.UnknownError	未知的内部错误。

调整同步任务规格

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

调整同步任务规格，此接口只支持按量计费任务的调整，调用此接口后不会立即生效，后台调整时间大概为3~5分钟。调用此接口后可通过查询同步任务信息接口DescribeSyncJobs，获取变配后的状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResizeSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-werwfs23
NewInstanceClass	是	String	任务规格 示例值：small

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 调整同步任务规格

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ResizeSyncJob
<公共请求参数>

{
"NewInstanceClass": "small",
"JobId": "sync-gb3l21wq"
}
```

输出示例

```
{
"Response": {
"RequestId": "ee736d91-eb62-4bd9-8dba-7e3c81350740"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.InternalErrorError	内部错误。
InternalError.InternalTradeError	调用计费服务失败。
InternalError.NotEnoughMoneyError	用户余额不足。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

修改同步任务配置

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

该接口支持在同步任务启动后修改任务的配置
修改同步配置的完整流程：修改同步任务配置->创建修改同步任务配置的校验任务->查询修改配置的校验任务的结果->启动修改配置任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySyncJobConfig。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-adasasww
DynamicObjects	否	Objects	修改后的同步对象
DynamicOptions	否	DynamicOptions	修改后的同步任务选项

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置修改对象任务

在需要修改对象时、通过该接口开始配置修改对象的任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifySyncJobConfig
<公共请求参数>

{
"JobId": "sync-4t8k7e8t",
"DynamicObjects": {
"Mode": "Partial",
"Databases": [
```

```
{
"DbName": "db1",
"DbMode": "All",
"TableMode": "All",
"ViewMode": "All"
}
]
},
"DynamicOptions": {
"ConflictHandleOption": {},
"ConflictHandleType": "ReportError",
"DdlOptions": [
{
"DdlObject": "Database",
"DdlValue": []
},
{
"DdlObject": "Table",
"DdlValue": []
},
{
"DdlObject": "View",
"DdlValue": []
},
{
"DdlObject": "Index",
"DdlValue": []
}
],
"OpTypes": [
"Insert",
"Update",
"Delete",
"DDL"
]
}
}
```

输出示例

```
{
"Response": {
"RequestId": "xxxaasassas"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied.OperationDeniedError	操作被拒绝。

创建修改同步配置的校验任务

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

在修改同步任务的配置后、通过该接口校验当前任务是否支持修改对象操作

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateModifyCheckSyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-cxsasdaq

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建修改对象的校验任务

在修改对象配置完成后、通过该接口创建修改对象的校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateModifyCheckSyncJob
<公共请求参数>

{
"JobId": "sync-xxasaas"
}
```

输出示例

```
{
"Response": {
"RequestId": "abcasasdasdasdas"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见 公共错误码。

查询修改对象的校验任务的结果

最近更新时间：2025-04-25 01:29:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

在创建修改对象的校验任务后、通过该接口查看校验任务的结果

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeModifyCheckSyncJobResult。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-csasweee

3. 输出参数

参数名称	类型	描述
Status	String	校验任务执行状态，如：notStarted(未开始)、running(校验中)、failed(校验任务失败)、success(任务成功) 示例值：success
StepCount	Integer	校验的步骤总数 示例值：4
StepCur	Integer	当前所在步骤 示例值：2
Progress	Integer	总体进度，范围为[0,100] 示例值：10
StepInfos	Array of StepInfo	步骤详细信息 注意：此字段可能返回 null，表示取不到有效值。
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询修改对象校验任务的结果

在创建修改对象校验任务后、查询校验任务的结果

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeModifyCheckSyncJobResult
```

```
<公共请求参数>

{
"JobId": "sync-xxsasa"
}
```

输出示例

```
{
"Response": {
"Progress": 100,
"RequestId": "00ba3080-f61d-11ed-926c-37ede6ab7ad5",
"Status": "success",
"StepCount": 15,
"StepCur": 15,
"StepInfos": [
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConnectDBCheck",
"StepName": "连接DB检查",
"StepNo": 1,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "OptimizeCheck",
"StepName": "周边检查",
"StepNo": 2,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "VersionCheck",
"StepName": "版本检查",
"StepNo": 3,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SrcPrivilegeCheck",
"StepName": "源实例权限检查",
"StepNo": 4,
"Warnings": [
{
```

```
"Code": "Warning: SrcPrivilegeCheck",
"HelpDoc": "",
"Message": "您的授权情况：GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE, CREATE ROLE, DROP ROLE ON *.* TO `root`@`%` WITH GRANT OPTION,GRANT APPLICATION_PASSWORD_ADMIN,ROLE_ADMIN,SHOW_ROUTINE,XA_RECOVER_ADMIN ON *.* TO `root`@`%` WITH GRANT OPTION。 源端账号权限范围大于要求的账号权限范围，腾讯云DTS不会使用超过必要的权限，请按照最小化原则给与授权：https://cloud.tencent.com/document/product/571/58700",
"SkipInfo": "",
"Solution": ""
}
]
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "SimpleParamCheck",
"StepName": "部分实例参数检查",
"StepNo": 5,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstPrivilegeCheck",
"StepName": "目标实例权限检查",
"StepNo": 6,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstEmptyCheck",
"StepName": "目标实例内容冲突检查",
"StepNo": 7,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "DstSpaceCheck",
"StepName": "目标实例空间检查",
"StepNo": 8,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
```

```
"Status": "finished",
"StepId": "BinlogParamCheck",
"StepName": "binlog参数检查",
"StepNo": 9,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintCheck",
"StepName": "外键依赖检查",
"StepNo": 10,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ConstraintRefCheck",
"StepName": "外键部分库表依赖检查",
"StepNo": 11,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ViewCheck",
"StepName": "视图检查",
"StepNo": 12,
"Warnings": []
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "WarningParamCheck",
"StepName": "警告项检查",
"StepNo": 13,
"Warnings": [
{
"Code": "Warn: WarningParamCheck",
"HelpDoc": "",
"Message": "对于既没有主键、也没有不能为null的唯一键的表，有数据重复的风险。 请参考 https://cloud.tencent.com/document/product/571/58
739",
"SkipInfo": "",
"Solution": ""
}
]
},
{
"Errors": [],
```

```
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "AdvancedStoreObjectCheck",
"StepName": "高级对象检查",
"StepNo": 14,
"Warnings": [
{
"Code": "Error: AdvancedStoreObjectCheck",
"HelpDoc": "",
"Message": "检测到进行了库表重命名，可能导致视图、函数，存储过程不能正常工作",
"SkipInfo": "",
"Solution": ""
}
]
},
{
"Errors": [],
"Progress": 100,
"StartTime": "",
"Status": "finished",
"StepId": "ModifyConfigCheck",
"StepName": "配置修改检查",
"StepNo": 15,
"Warnings": []
}
]
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameter.InvalidParameterError	参数无效。

开始修改配置流程

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

在查询修改对象的校验任务的结果中的status为success后、通过该接口开始修改配置流程

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartModifySyncJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	同步任务id 示例值：sync-csasssww

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 开始修改对象

在修改对象校验通过后、通过该接口开始修改对象

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StartModifySyncJob
<公共请求参数>

{
"JobId": "sync-sxxssss"
}
```

输出示例

```
{
"Response": {
"RequestId": "abcsasxasxasxasxa"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameter.InvalidParameterError	参数无效。
ResourceNotFound.ResourceNotFoundError	资源未找到。

修改同步任务的传输速率

最近更新时间：2025-04-25 01:29:22

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

用户在发现同步任务对用户的数据库的负载影响较大时、可通过该接口限制任务的传输速率

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySyncRateLimit。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务ID 示例值：dts-saasxa
DumpThread	否	Integer	同步任务全量导出线程数、有效值为 1-16 示例值：8
DumpRps	否	Integer	同步任务全量导出的 Rps 限制、需要大于 0 示例值：20000
LoadThread	否	Integer	同步任务全量导入线程数、有效值为 1-16 示例值：8
SinkerThread	否	Integer	同步任务增量导入线程数、有效值为 1-128 示例值：32
LoadRps	否	Integer	同步任务全量导入的Rps 示例值：8

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改任务全量导出线程数

修改任务全量导出线程数成功

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifySyncRateLimit
```

```
<公共请求参数>

{
"JobId": "sync-asasdasa",
"DumpThread": 12,
"DumpRps": 0,
"LoadThread": 0,
"SinkerThread": 0,
"LoadRps": 0
}
```

输出示例

```
{
"Response": {
"RequestId": "abc"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
ResourceNotFound.ResourceNotFoundError	资源未找到。

数据一致性校验相关接口
查询一致性校验任务详情

最近更新时间：2025-04-25 01:29:25

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询一致性校验任务详情

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeCompareReport。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-amm1jw5q
CompareTaskId	是	String	校验任务 Id 示例值：dts-amm1jw5q-cmp-bmuum7jk
DifferenceLimit	否	Integer	校验不一致结果的 limit 示例值：10
DifferenceOffset	否	Integer	不一致的 Offset 示例值：0
DifferenceDB	否	String	搜索条件，不一致的库名 示例值：db1
DifferenceTable	否	String	搜索条件，不一致的表名 示例值：t1
SkippedLimit	否	Integer	未校验的 Limit 示例值：10
SkippedOffset	否	Integer	未校验的 Offset 示例值：0
SkippedDB	否	String	搜索条件，未校验的库名 示例值：db1
SkippedTable	否	String	搜索条件，未校验的表名 示例值：t1

3. 输出参数

参数名称	类型	描述
Abstract	CompareAbstractInfo	一致性校验摘要信息
Detail	CompareDetailInfo	一致性校验详细信息

版权所有：腾讯云计算（北京）有限责任公司

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询一致性校验任务详情

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCompareReport
<公共请求参数>

{
"JobId": "dts-amm1jw5q",
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk"
}
```

输出示例

```
{
"Response": {
"Abstract": {
"CheckedTables": 1,
"Conclusion": "same",
"DifferentRows": 0,
"DifferentTables": 0,
"DstSampleRows": 0,
"FinishedAt": "2024-12-20 19:07:20",
"NearlyTableCount": 1,
"Objects": {
"AdvancedObjects": [],
"ObjectItems": [
{
"DbMode": "partial",
"DbName": "db_1",
"SchemaName": "",
"TableMode": "partial",
"Tables": [
{
"ColumnMode": "all",
"Columns": [],
"TableName": "tb_1"
}
],
"ViewMode": "partial",
"Views": null
}
],
"ObjectMode": "partial"
},
"Options": {
"Method": "dataCheck",
```

```
"SampleRate": 100,
"ThreadCount": 1
},
"SkippedTables": 0,
"SrcSampleRows": 0,
"StartedAt": "2024-12-20 19:04:03",
"Status": "success",
"TotalTables": 1
},
"Detail": {
"Difference": {
"Items": [],
"TotalCount": 0
},
"DifferenceAdvancedObjects": {
"Items": null,
"TotalCount": 0
},
"DifferenceData": {
"Items": null,
"TotalCount": 0
},
"DifferenceRow": {
"Items": null,
"TotalCount": 0
},
"Skipped": {
"Items": [],
"TotalCount": 0
}
},
"RequestId": "a3365b29-2042-4565-9206-8df9e2af1a17"
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

终止一致性校验任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

终止一致性校验任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StopCompare。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-amm1jw5q
CompareTaskId	是	String	对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-8yv4w2i1-cmp-37skmii9
ForceStop	否	Boolean	是否强制停止。如果填true，同步任务增量阶段会跳过一致性校验产生的binlog，达到快速恢复任务的效果 示例值：true

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 终止一致性校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StopCompare
<公共请求参数>

{
"JobId": "dts-amm1jw5q",
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

删除一致性校验任务

最近更新时间：2025-04-25 01:29:25

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

删除一致性校验任务。当一致性校验任务状态为success、failed、canceled 时可以执行此操作。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DeleteCompareTask。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-8yv4w2i1
CompareTaskId	是	String	对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-8yv4w2i1-cmp-37skmii9

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除一致性校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DeleteCompareTask
<公共请求参数>

{
"JobId": "dts-amm1jw5q",
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValue	参数值错误。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。

错误码	描述
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改一致性校验任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改一致性校验任务，在任务创建后启动之前，可修改一致性校验参数

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyCompareTask。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务 Id 示例值：dts-45jhpmjh
CompareTaskId	是	String	对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-45jhpmjh-cmp-k7uhq1v3
TaskName	否	String	任务名称 示例值：初始校验任务
ObjectMode	否	String	数据对比对象模式，sameAsMigrate(全部迁移对象， 默认为此项配置)、custom(自定义)，注意自定义对比对象必须是迁移对象的子集 示例值：sameAsMigrate
Objects	否	CompareObject	对比对象，若CompareObjectMode取值为custom，则此项必填
Options	否	CompareOptions	一致性校验选项

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改一致性校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyCompareTask
<公共请求参数>
```

```
{
"JobId": "dts-p1sposne",
"CompareTaskId": "dts-8yv4w2i1-cmp-37skmii9",
"TaskName": "test_cmp",
"ObjectMode": "custom",
"Objects": {
"ObjectMode": "partial",
"ObjectItems": [
{
"DbName": "big100",
"DbMode": "partial",
"TableMode": "partial",
"Tables": [
{
"TableName": "sbtest1"
}
]
},
{
"DbName": "db1",
"DbMode": "all",
"TableMode": "all",
"ViewMode": "all"
}
]
}
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改一致性校验任务名称

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改一致性校验任务名称

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyCompareTaskName。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-8yv4w2i1
CompareTaskId	是	String	对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-8yv4w2i1-cmp-37skmii9
TaskName	是	String	一致性校验任务名称 示例值：cmp_test

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改一致性校验任务名称

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyCompareTaskName
<公共请求参数>

{
"JobId": "dts-p1sposne",
"TaskName": "test_cmp",
"CompareTaskId": "dts-p1sposne-cmp-37skmii9"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

创建一致性校验任务

最近更新时间：2025-04-25 01:29:25

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于创建数据对比任务，创建成功后会返回数据对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9，创建成功后可通过StartCompare启动一致性校验任务

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateCompareTask。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	任务 Id 示例值：dts-p1sposne
TaskName	否	String	数据对比任务名称，若为空则默认给CompareTaskId相同值 示例值：compare_test
ObjectMode	否	String	数据对比对象模式，sameAsMigrate(全部迁移对象， 默认为此项配置)，custom(自定义模式) 示例值：sameAsMigrate
Objects	否	CompareObject	数据对比对象，当ObjectMode为custom时，此项需要填写。
Options	否	CompareOptions	一致性校验选项

3. 输出参数

参数名称	类型	描述
CompareTaskId	String	数据对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-8yv4w2i1-cmp-37skmii9
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建一致性校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateCompareTask
<公共请求参数>
```

```
{
"JobId": "dts-p1sposne",
"TaskName": "test_cmp",
"ObjectMode": "custom",
"Objects": {
"ObjectMode": "partial",
"ObjectItems": [
{
"DbName": "big100",
"DbMode": "partial",
"TableMode": "partial",
"Tables": [
{
"TableName": "sbtest1"
}
]
},
{
"DbName": "db1",
"DbMode": "all",
"TableMode": "all",
"ViewMode": "all"
}
]
}
}
```

输出示例

```
{
"Response": {
"CompareTaskId": "dts-8yv4w2i1-cmp-37skmii9",
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InternalError.InternalHttpServerError	http请求访问出错。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

启动一致性校验任务

最近更新时间：2025-04-25 01:29:24

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

启动一致性校验任务，启动之前需要先通过接口 `CreateCompareTask` 创建一致性校验任务，启动后可通过接口 `DescribeCompareTasks` 查询一致性校验任务列表来获得启动后的状态

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartCompare。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-8yv4w2i1
CompareTaskId	是	String	对比任务 ID，形如：dts-8yv4w2i1-cmp-37skmii9 示例值：dts-8yv4w2i1-cmp-37skmii9

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 启动一致性校验任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StartCompare
<公共请求参数>

{
"JobId": "dts-1i9hna5e",
"CompareTaskId": "dts-1i9hna5e-cmp-kjqrl13s"
}
```

输出示例

```
{
"Response": {
"RequestId": "ac300ff0-00f2-11ed-b005-4930e69d89c2"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

查询一致性校验任务列表

最近更新时间：2025-04-25 01:29:25

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询一致性校验任务列表，调用该接口后可通过接口 `DescribeCompareTasks` 查询一致性校验任务列表来获得启动后的状态。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeCompareTasks。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	迁移任务 Id 示例值：dts-e7ukka6g
Limit	否	Integer	分页设置，表示每页显示多少条任务，默认为 20 示例值：20
Offset	否	Integer	分页偏移量 示例值：0
CompareTaskId	否	String	校验任务 ID 示例值：dts-5ftw8366-compare-n0pee6pc
Status.N	否	Array of String	任务状态过滤，可能的值：created - 创建完成；readyRun - 等待运行；running - 运行中；success - 成功；stopping - 结束中；failed - 失败；canceled - 已终止 示例值：["running"]

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	数量 示例值：1
Items	Array of CompareTaskItem	一致性校验列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询一致性校验任务列表

查询一致性校验任务列表

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeCompareTasks
<公共请求参数>

{
"JobId": "dts-e7ukka6g"
}
```

输出示例


```
{
"Response": {
"Items": [
{
"CheckProcess": {
"Message": "",
"Status": "",
"StepAll": 0,
"StepNow": 0,
"Percent": 0,
"Steps": []
},
"CompareProcess": {
"Message": "done",
"Status": "",
"StepAll": 0,
"StepNow": 0,
"Steps": []
},
"CompareTaskId": "dts-amm1jw5q-cmp-bmuum7jk",
"Conclusion": "same",
"Config": {
"ObjectItems": [
{
"DbMode": "partial",
"DbName": "big100",
"SchemaName": "",
"TableMode": "partial",
"Tables": null,
"ViewMode": "none",
"Views": null
},
{
"DbMode": "all",
"DbName": "db1",
"SchemaName": "",
"TableMode": "all",
"Tables": null,
"ViewMode": "all",
"Views": null
}
],
"ObjectMode": "partial",
"AdvancedObjects": [
```

```
"index"
]
},
"CreatedAt": "2022-07-11 17:21:02",
"FinishedAt": "2022-07-11 17:24:54",
"Method": "dataCheck",
"Options": {
"Method": "dataCheck",
"SampleRate": 100,
"ThreadCount": 1
},
"JobId": "dts-amm1jw5q",
"Message": "done",
"StartedAt": "2022-07-11 17:24:09",
"Status": "success",
"TaskName": "defaultCmpTask"
}
],
"RequestId": "11d35c90-01bb-11ed-bad9-7b3bbe11abda",
"TotalCount": 1
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
InternalError.InternalErrorError	内部错误。

错误码	描述
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

（新版）数据订阅相关接口
创建订阅任务

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(CreateSubscribe)用于创建一个数据订阅任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
Product	是	String	订阅的数据库类型，目前支持 cynosdbmysql(tdsql-c mysql版),mariadb,mongodb,mysql,percona,tdpg(tdsql postgresql版),tdsqlpercona(tdsql mysql版) 示例值：mysql
PayType	是	Integer	付费方式，枚举值：0-包年包月，1-按量计费 示例值：0
Duration	否	Integer	购买时长。当 payType 为包年包月时，该项需要填，单位为月，最小值为 1，最大值为 120。不填默认1 示例值：1
AutoRenew	否	Integer	是否自动续费。当 payType 为包年包月时，该项需要填。枚举值：0-不自动续费，1-自动续费。默认不自动续费。按量计费设置该标识无效。 示例值：0
Count	否	Integer	购买数量,默认为1，最大为10 示例值：1
Tags.N	否	Array of TagItem	实例资源标签
Name	否	String	任务名，自定义 示例值：binlog订阅

3. 输出参数

参数名称	类型	描述
SubscribeIds	Array of String	数据订阅实例的ID数组 示例值：["subs-b6x64o31tm"]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 购买订阅任务

购买1个按量计费的mysql订阅任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateSubscribe
<公共请求参数>

{
"Product": "mysql",
"PayType": 1,
"Count": 1,
"Tags": [
{
"TagKey": "负责人",
"TagValue": "jason"
}
],
"Name": "binlog订阅"
}
```

输出示例

```
{
"Response": {
"SubscribeIds": [
"subs-b6x64o31tm"
],
"RequestId": "xxxxxxxxxxxxxxx"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.UserNotAuthed	用户没有实名认证。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.NotEnoughMoneyError	用户余额不足。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

配置订阅任务

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ConfigureSubscribeJob)用于配置数据订阅实例。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ConfigureSubscribeJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-xxxxxx
SubscribeMode	是	String	数据订阅的类型，当 DatabaseType 不为 mongodb 时，枚举值为：all-全实例更新；dml-数据更新；ddl-结构更新；dmlAndDdl-数据更新+结构更新。当 DatabaseType 为 mongodb 时，枚举值为 all-全实例更新；database-订阅单库；collection-订阅单集合 示例值：dmlAndDdl
AccessType	是	String	源数据库接入类型，如：extranet(公网)、vpncloud(vpn接入)、dcg(专线接入)、ccn(云联网)、cdb(云数据库)、cvm(云服务器自建)、intranet(自研上云)、vpc(私有网络vpc)。注意具体可选值依赖当前链路支持能力 示例值：cdb
Endpoints.N	是	Array of EndpointItem	数据库节点信息
KafkaConfig	是	SubscribeKafkaConfig	Kafka配置
SubscribeObjects.N	否	Array of SubscribeObject	订阅的数据库表信息，当 SubscribeMode 不为 all和ddl 时，SubscribeObjects 为必选参数
Protocol	否	String	订阅数据格式，如：protobuf、json、avro。注意具体可选值依赖当前链路支持能力，数据格式详情参考官网的消费demo文档 示例值：protobuf
PipelineInfo.N	否	Array of PipelineInfo	mongo选填参数：输出聚合设置。
ExtraAttr.N	否	Array of KeyValuePairOption	为业务添加的额外信息。参数名作key，参数值作value。 mysql选填参数：ProcessXA-是否处理XA事务，填true处理，不填或填其他值不处理。 mongo选填参数：SubscribeType-订阅类型，目前只支持changeStream，不填也是默认changeStream。 其他业务暂没有可选参数。

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置订阅

配置订阅一个cdb整实例的任务，其中db1.table1的数据根据id列值投递到不同的kafka分区，其他的数据根据表名投递

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ConfigureSubscribeJob
<公共请求参数>

{
"SubscribeMode": "all",
"SubscribeId": "subs-5ft0e2nrc0",
"KafkaConfig": {
"DistributeRules": [
{
"DbPattern": "db1",
"RuleType": "cols",
"TablePattern": "table1",
"Columns": [
"id"
]
}
],
"NumberOfPartitions": 8,
"DefaultRuleType": "table"
},
"AccessType": "cdb",
"Protocol": "json",
"Endpoints": [
{
"InstanceId": "cdb-kdxona7h",
"DatabaseRegion": "ap-guangzhou",
"EncryptConn": "Encrypted",
"User": "root",
"Password": "******"
}
],
"ExtraAttr": [
{
"Value": "true",
"Key": "ProcessXA"
}
]
}
```

输出示例

```
{
"Response": {
"RequestId": "abc"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.ControllerNotFoundError	异常错误导致当前接口未注册。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

校验订阅任务

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(CreateSubscribeCheckJob)用于创建一个订阅校验任务。任务必须已经成功调用ConfigureSubscribeJob接口配置了所有的必要信息才能启动校验。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateSubscribeCheckJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-xxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 启动一个订阅任务的校验

启动一个订阅任务的校验。任务必须已经成功调用ConfigureSubscribe接口配置了所有的必要信息才能启动校验。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateSubscribeCheckJob
<公共请求参数>

{
"SubscribeId": "subs-l4d3a7izai"
}
```

输出示例

```
{
"Response": {
"RequestId": "77501ba0-f097-11ed-ac7d-b1056735c0c8"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

查询订阅校验结果

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeSubscribeCheckJob)用于查询订阅校验任务结果。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribeCheckJob。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-l4d3a7izai

3. 输出参数

参数名称	类型	描述
SubscribeId	String	订阅实例ID 示例值：subs-xxxxxx
Message	String	失败或者报错提示，成功则提示success。 示例值：success
Status	String	任务运行状态，可能值为 running,failed,success 示例值：success
Progress	Integer	当前总体进度，范围 0~100 示例值：100
StepAll	Integer	校验总步骤数 示例值：5
StepNow	Integer	当前执行步骤 示例值：5
Steps	Array of SubscribeCheckStepInfo	各个步骤运行状态
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询订阅校验任务结果

查询订阅校验任务结果

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeSubscribeCheckJob
<公共请求参数>

{
"SubscribeId": "subs-l4d3a7izai"
}
```

输出示例

```
{
"Response": {
"Message": "success",
"Progress": 100,
"RequestId": "b048a800-f2fe-11ed-a211-5df19a912eab",
"Status": "success",
"StepAll": 5,
"StepNow": 5,
"Steps": [
{
"Errors": [],
"Percent": 100,
"Status": "finished",
"StepId": "ConnectDBCheck",
"StepName": "连接DB检查",
"StepNo": 1,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"Status": "finished",
"StepId": "OptimizeCheck",
"StepName": "周边检查",
"StepNo": 2,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"Status": "finished",
"StepId": "VersionCheck",
"StepName": "版本检查",
"StepNo": 3,
"Warnings": []
},
{
"Errors": [],
"Percent": 100,
"Status": "finished",
"StepId": "SrcPrivilegeCheck",
"StepName": "源实例权限检查",
"StepNo": 4,
"Warnings": [
```

```
{
"HelpDoc": "",
"Message": "我们不需要能修改您源端数据的权限，请取消他们。例如，您的授权情况：GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, REL
OAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE,
REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE,
CREATE ROLE, DROP ROLE ON *.* TO `root`@`%` WITH GRANT OPTION,GRANT APPLICATION_PASSWORD_ADMIN,ROLE_ADMIN,SHOW_ROUTINE,XA_
RECOVER_ADMIN ON *.* TO `root`@`%` WITH GRANT OPTION 请参考 https://cloud.tencent.com/document/product/571/52412"
}
]
},
{
"Errors": [],
"Percent": 100,
"Status": "finished",
"StepId": "BinlogParamCheck",
"StepName": "binlog参数检查",
"StepNo": 5,
"Warnings": []
}
],
"SubscribeId": "subs-9jyki7hniw"
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。

错误码	描述
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

启动订阅任务

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(StartSubscribe)用于启动一个kafka版本的数据订阅实例。只有当订阅任务的状态为checkPass时，才能调用本接口。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-xxxxxx

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 开启数据订阅任务

开启数据订阅任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: StartSubscribe
<公共请求参数>

{
"SubscribeId": "subs-635ns8r71g"
}
```

输出示例

```
{
"Response": {
"RequestId": "da826d80-f2fa-11ed-9136-c7bfacc9cf90"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。

错误码	描述
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

修改订阅对象

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeObjects)用于修改数据订阅对象和kafka分区规则，如果是mongo订阅，还可以修改输出聚合规则。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeObjects。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-9jyki7hniw
SubscribeObjectType	是	Integer	数据订阅的类型，非mongo任务的枚举值：0-全实例更新；1-数据更新；2-结构更新；3-数据更新+结构更新。mongo任务的枚举值：0-全实例更新；4-订阅单库；5-订阅单集合 示例值：0
Objects.N	否	Array of ModifiedSubscribeObject	修改后的订阅数据库表信息。会覆盖原来的订阅对象，所以除非SubscribeObjectType = 0或2，否则改字段必填。
DistributeRules.N	否	Array of DistributeRule	kafka分区策略。如果不填，默认不修改。如果填了，会覆盖原来的策略。
DefaultRuleType	否	String	默认分区策略。不满足DistributeRules中正则表达式的数据，将按照默认分区策略计算分区。 非mongo产品支持的默认策略: table-按表名分区，pk-按表名+主键分区。mongo的默认策略仅支持：collection-按集合名分区。 该字段与DistributeRules搭配使用。如果配置了DistributeRules，该字段也必填。 如果配置了该字段，视为配置了一条DistributeRules，原来的分区策略也会被覆盖。 示例值：table
PipelineInfo.N	否	Array of PipelineInfo	mongo输出聚合设置，mongo任务可选。如果不填，默认不修改。

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅通道的订阅规则

修改数据订阅的订阅规则，如添加或者移除某些库表

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifySubscribeObjects
&SubscribeId=subs-ieuwi83j2e
&SubscribeObjectType=0
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

查询订阅任务详情

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeSubscribeDetail)获取数据订阅实例的配置信息。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribeDetail。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID 示例值：subs-xxxxxx

3. 输出参数

参数名称	类型	描述
SubscribeId	String	数据订阅的ID，形如subs-b6x64o31tm 示例值：subs-xxxxxx
SubscribeName	String	数据订阅实例的名称 示例值：binlog订阅
Product	String	订阅的数据库类型，目前支持 cynosdbmysql(tdsql-c mysql版),mariadb,mongodb,mysql,percona,tdpg(tdsql postgresql版),tdsqlpercona(tdsql mysql版) 示例值：mysql
InstanceId	String	订阅的云数据库实例ID，只有订阅云数据库该值才有意义 示例值：cdb-xxxxxx
InstanceStatus	String	订阅的云数据库实例状态，只有订阅云数据库该值才有意义。可能值为：running, isolated, offline 示例值：running
Status	String	订阅任务计费状态，可能值为：正常normal, 隔离中isolating, 已隔离isolated, 下线中offlining, 按量转包年包月中 post2PrePayIng 示例值：normal
SubsStatus	String	订阅任务状态，可能值为：未启动notStarted, 校验中checking, 校验不通过checkNotPass, 校验通过checkPass, 启动中starting, 运行中running, 异常出错error 示例值：running
ModifyTime	String	修改时间，时间格式如：Y-m-d h:m:s 示例值：2023-06-30 12:00:00
CreateTime	String	创建时间，时间格式如：Y-m-d h:m:s 示例值：2023-06-30 12:00:00

参数名称	类型	描述
IsolateTime	String	隔离时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：0000-00-00 00:00:00
ExpireTime	String	包年包月任务的到期时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：2023-07-30 12:00:00
OfflineTime	String	下线时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：0000-00-00 00:00:00
PayType	Integer	付费方式，可能值为：0-包年包月，1-按量计费 示例值：0
AutoRenewFlag	Integer	自动续费标识。只有当 PayType=0，该值才有意义。枚举值：0-不自动续费，1-自动续费 示例值：1
Region	String	任务所在地域 示例值：ap-guangzhou
Topic	String	Kafka topic 示例值：topic-subs-9n0q6o7tok-cdb-ixunfa27
Broker	String	Kafka服务Broker地址 示例值：guangzhoutest-kafka-5.cdb-dts.tencentcs.com.cn:32339
SubscribeMode	String	数据订阅的类型，当 Product 不为 mongodb 时，可能值为：all-全实例更新；dml-数据更新；ddl-结构更新；dmlAndDdl-数据更新+结构更新。当 Product 为 mongodb 时，可能值为 all-全实例更新；database-订阅单库；collection-订阅单集合 示例值：dmlAndDdl
Protocol	String	订阅数据格式。如果为空则用的默认格式: mysql\cynosdbmysql\mariadb\percona\tdsqlpercona\tdpg是protobuf，mongo是json。当 DatabaseType 为 mysql和cynosdbmysql 时有三种可选协议：protobuf\avro\json。数据格式详情参考官网的消费demo文档 示例值：protobuf
SubscribeObjects	Array of SubscribeObject	订阅的数据库表信息
KafkaConfig	SubscribeKafkaConfig	kafka配置信息
KafkaVersion	String	订阅内置kafka的版本信息 示例值：2.6.0
AccessType	String	源数据库接入类型，如：extranet(公网)、vpncloud(vpn接入)、dcg(专线接入)、ccn(云联网)、cdb(云数据库)、cvm(云服务器自建)、intranet(自研上云)、vpc(私有网络vpc)。注意具体可选值依赖当前链路支持能力 示例值：cdb
Endpoints	Array of EndpointItem	接入类型信息
PipelineInfo	Array of PipelineInfo	mongo输出聚合设置
Tags	Array of TagItem	标签
Errors	Array of SubsErr	订阅任务报错信息 注意：此字段可能返回 null，表示取不到有效值。
ExtraAttr	Array of KeyValuePairOption	为业务添加的额外信息。参数名作key，参数值作value。 mysql选填参数：ProcessXA-是否处理XA事务，为true处理，其他不处理。 mongo选填参数：SubscribeType-订阅类型，目前只支持changeStream。
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询订阅任务详情

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeSubscribeDetail
<公共请求参数>

{
"SubscribeId": "subs-0b2up6hk4u"
}
```

输出示例

```
{
"Response": {
"Region": "ap-guangzhou",
"AccessType": "cdb",
"AutoRenewFlag": 0,
"Broker": "guangzhoutest-kafka-3.cdb-dts.tencentcs.com.cn:32339",
"CreateTime": "2023-12-06 14:31:27",
"Endpoints": [
{
"DatabaseRegion": "ap-guangzhou",
"EncryptConn": "UnEncrypted",
"InstanceId": "cdb-kdxona7h",
"User": "root"
}
],
"ExpireTime": "0000-00-00 00:00:00",
"InstanceId": "cdb-kdxona7h",
"InstanceStatus": "running",
"IsolateTime": "0000-00-00 00:00:00",
"KafkaConfig": {
"DistributeRules": [
{
"DbPattern": "db1",
"RuleType": "cols",
"TablePattern": "table1",
"Columns": [
"id"
]
},
{
"DbPattern": ".*",
"RuleType": "table",
"TablePattern": ".*",
"Columns": null
}
],
"NumberOfPartitions": 8
},
"ModifyTime": "2023-12-06 14:33:47",
"OfflineTime": "0000-00-00 00:00:00",
"PayType": 0,
"Product": "mysql",
```

```
"Protocol": "json",
"Status": "normal",
"SubsStatus": "running",
"SubscribeId": "subs-0b2up6hk4u",
"SubscribeName": "binlog订阅",
"SubscribeMode": "all",
"Topic": "topic-subs-0b2up6hk4u-cdb-kdxona7h",
"RequestId": "7bd11a40-9406-11ee-9689-216abbbe2107"
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。

错误码	描述
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

获取数据订阅实例列表

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeSubscribes)获取数据订阅实例信息列表，默认分页，每次返回20条

默认接口请求频率限制：100次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribeJobs。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	否	String	订阅 ID 筛选，精确匹配 示例值：subs-9jyki7hniw
SubscribeIds.N	否	Array of String	订阅 ID 筛选，精确匹配 示例值：["subs-9jyki7hniw","subs-31adyjj41f"]"]
SubscribeName	否	String	订阅名称，前缀模糊匹配 示例值：binlog订阅
InstanceId	否	String	订阅的云上数据库实例的 ID，精确匹配 示例值：cdb-kdxona7h
Topic	否	String	订阅的topicName 示例值：topic-subs-g9k9ct7gmc--6itdltb8
PayType	否	Integer	计费模式筛选，可能的值：0-包年包月，1-按量计费 示例值：1
Product	否	String	订阅的数据库产品，目前支持 cynosdbmysql,mariadb,mongodb,mysql,percona,tdpg,tdsqlpercona(tdsqlmysql) 示例值：mysql
Status.N	否	Array of String	数据订阅生命周期状态，可能的值为：正常 normal, 隔离中 isolating, 已隔离 isolated, 下线中 offlining，按量转包年包月中 post2PrePayIng 示例值：normal
SubsStatus.N	否	Array of String	数据订阅状态，可能的值为：未启动 notStarted, 校验中 checking, 校验不通过 checkNotPass, 校验通过 checkPass, 启动中 starting, 运行中 running, 异常出错 error 示例值：running
Offset	否	Integer	返回记录的起始偏移量。默认0 示例值：0
Limit	否	Integer	单次返回的记录数量。默认20，最大100 示例值：20
OrderDirection	否	String	排序方向，可选的值为"DESC"和"ASC"，默认为"DESC"，按创建时间逆序排序 示例值：ASC
TagFilters.N	否	Array of	tag 过滤条件，多个 TagFilter 之间关系为且

参数名称	必选	类型	描述
		TagFilter	

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	符合查询条件的实例总数 示例值：1
Items	Array of SubscribeInfo	数据订阅实例的信息列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询订阅任务列表

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeSubscribeJobs
<公共请求参数>

{
"Offset": 0,
"Limit": 10
}
```

输出示例

```
{
"Response": {
"TotalCount": 1,
"Items": [
{
"SubscribeId": "subs-5ft0e2nrc0",
"SubscribeName": "name-5ft0e2nrc0",
"Topic": "topic-subs-5ft0e2nrc0-cdb-kdxona7h",
"Product": "mysql",
"InstanceId": "cdb-kdxona7h",
"InstanceStatus": "running",
"Status": "normal",
"SubsStatus": "running",
"ModifyTime": "2023-10-01 12:00:00",
"CreateTime": "2023-09-01 12:00:00",
"IsolateTime": "0000-00-00 00:00:00",
"ExpireTime": "2023-11-01 12:00:00",
"OfflineTime": "0000-00-00 00:00:00",
"PayType": 0,
"AutoRenewFlag": 0,
"Region": "ap-guangzhou",
"AccessType": "cdb",
```

```
"Endpoints": [
{
"DatabaseRegion": "ap-guangzhou",
"InstanceId": "cdb-kdxona7h",
"EncryptConn": "Encrypted",
"User": "root"
}
],
"SubscribeVersion": "kafka"
}
],
"RequestId": "cd6eebd0-9405-11ee-9689-216abbe2107"
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。

错误码	描述
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

重试订阅任务

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ResumeSubscribe) 用于恢复报错的订阅任务。当订阅任务的状态为error时，可通过本接口尝试对任务进行恢复。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResumeSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-9jyki7hniw

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 恢复订阅任务

通过重启尝试恢复一个报错的任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ResumeSubscribe
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw"
}
```

输出示例

```
{
"Response": {
"RequestId": "b048a800-f2fe-11ed-a211-5df19a912eab"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。

错误码	描述
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

创建消费组

最近更新时间：2025-04-25 01:29:21

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

为订阅实例创建消费者组

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateConsumerGroup。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例id 示例值：subs-jvxbbos7c0
ConsumerGroupName	是	String	消费组名称，以数字、字母(大小写)或者_ - .开头，以数字、字母(大小写)结尾。实际生成的消费组全称形如：consumer-grp-#{SubscribeId}-#{ConsumerGroupName} 示例值：group-1
AccountName	是	String	账号名称。以数字、字母(大小写)或者_ - .开头，以数字、字母(大小写)结尾。实际生成的账户全称形如：account-#{SubscribeId}-#{AccountName} 示例值：jason
Password	是	String	消费组密码，长度必须大于3 示例值：qJ3e1Wd~
Description	否	String	消费组备注 示例值：binlog订阅

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建消费组

创建消费组

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateConsumerGroup
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw",
"ConsumerGroupName": "group-1",
"AccountName": "jason",
```

```
"Password": "qJ3e1Wd~",
"Description": "this is a test account"
}
```

输出示例

```
{
"Response": {
"RequestId": "20461920-b18e-11ec-ae1a-cfe224f4f21f"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValue	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。

错误码	描述
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

获取数据订阅的消费组详情

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeConsumerGroups)用于获取订阅实例配置的消费者组详情。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeConsumerGroups。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例id 示例值：subs-47damshnra
Offset	否	Integer	返回记录的起始偏移量。默认0 示例值：0
Limit	否	Integer	单次返回的记录数量。默认10 示例值：1

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	指定实例下的消费者组总数 示例值：1
Items	Array of GroupInfo	消费者组列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取数据订阅的消费组

获取数据订阅的消费组

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeConsumerGroups
<公共请求参数>
```

```
{
"SubscribeId": "subs-9jyki7hniw"
}
```

输出示例

```
{
"Response": {
"Items": [
{
"Account": "account-subs-47damshnra-3",
"ConsumerGroupLag": 464948,
"ConsumerGroupName": "consumer-grp-subs-47damshnra-3",
"ConsumerGroupOffset": -1,
"ConsumerGroupState": "Dead",
"CreatedAt": "2022-03-25 17:31:31",
"Description": "sdfjo",
"Latency": 528268,
"PartitionAssignment": [],
"StateOfPartition": [
{
"ConsumerGroupLag": 464948,
"ConsumerGroupOffset": -1,
"Latency": 528268,
"PartitionNo": 0
}
],
"UpdatedAt": "2022-03-25 17:31:31"
},
{
"Account": "account-subs-47damshnra-2",
"ConsumerGroupLag": 464948,
"ConsumerGroupName": "consumer-grp-subs-47damshnra-2",
"ConsumerGroupOffset": -1,
"ConsumerGroupState": "Dead",
"CreatedAt": "2022-03-22 17:34:31",
"Description": "sdfjo",
"Latency": 528268,
"PartitionAssignment": [],
"StateOfPartition": [
{
"ConsumerGroupLag": 464948,
"ConsumerGroupOffset": -1,
"Latency": 528268,
"PartitionNo": 0
}
],
"UpdatedAt": "2022-03-22 17:34:31"
},
{
"Account": "account-subs-47damshnra-1",
"ConsumerGroupLag": 464948,
"ConsumerGroupName": "consumer-grp-subs-47damshnra-1",
"ConsumerGroupOffset": -1,
"ConsumerGroupState": "Dead",
"CreatedAt": "2022-03-04 15:14:18",
```

```
"Description": "sdfjo",
"Latency": 528268,
"PartitionAssignment": [],
"StateOfPartition": [
{
"ConsumerGroupLag": 464948,
"ConsumerGroupOffset": -1,
"Latency": 528268,
"PartitionNo": 0
}
],
"UpdatedAt": "2022-03-04 15:14:18"
}
],
"RequestId": "68c04ba0-b0d8-11ec-98f5-11f9ac4a2e37",
"TotalCount": 3
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。

错误码	描述
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改消费组备注

最近更新时间：2025-04-25 01:29:19

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifyConsumerGroupDescription)用于修改指定订阅消费组备注。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyConsumerGroupDescription。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-9jyki7hniw
ConsumerGroupName	是	String	消费组名称。实际的消费组全称形如：consumer-grp-#{SubscribeId}-#{ConsumerGroupName}。请务必保证消费组名称正确。 示例值：consumer-grp-subs-9jyki7hniw-group-1
AccountName	是	String	账户名称。实际的账户全称形如：account-#{SubscribeId}-#{AccountName}。请务必保证账户名称正确。 示例值：account-subs-9jyki7hniw-jason
Description	是	String	修改之后的消费组描述 示例值：binlog订阅

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改消费组备注

修改消费组备注

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyConsumerGroupDescription
<公共请求参数>

{
```

```
"SubscribeId": "subs-9jyki7hniw",
"AccountName": "account-subs-9jyki7hniw-1",
"ConsumerGroupName": "consumer-grp-subs-9jyki7hniw-1",
"Description": "this is a test group"
}
```

输出示例

```
{
"Response": {
"RequestId": "bef6a6f0-f3b1-11ed-b987-7301347aa36f"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。

错误码	描述
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改消费组密码

最近更新时间：2025-04-29 01:31:23

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifyConsumerGroupPassword)用于修改指定订阅消费组密码。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyConsumerGroupPassword。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-9jyki7hniw
AccountName	是	String	账号名称。实际的账户全称形如：account-#{SubscribeId}-#{AccountName} 示例值：account-subs-9jyki7hniw-jason
ConsumerGroupName	是	String	消费组名称。实际的消费组全称形如：consumer-grp-#{SubscribeId}-#{ConsumerGroupName} 示例值：consumer-grp-subs-9jyki7hniw-group-1
NewPassword	是	String	新密码。字符长度不小于3，不大于32 示例值：dliasyud

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改消费组密码

修改消费组密码

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifyConsumerGroupPassword
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw",
```

```
"AccountName": "account-subs-9jyki7hniw-group-1",
"ConsumerGroupName": "consumer-grp-subs-9jyki7hniw-jason",
"NewPassword": "asdiouy09"
}
```

输出示例

```
{
"Response": {
"RequestId": "bef6a6f0-f3b1-11ed-b987-7301347aa36f"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。

错误码	描述
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

查询kafka offset

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeOffsetByTime)查询KafkaTopic中指定时间前最近的offset。
接口输出的offset是离这个时间最近的offset。
如果输入时间比当前时间晚的多，相当于输出的就是最新的offset；
如果输入时间比当前时间早的多，相当于输出的就是最老的offset；
如果输入空，默认0时间，也就是查询最老的offset。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeOffsetByTime。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-9jyki7hniw
Time	是	String	时间点，格式为：Y-m-d h:m:s。如果输入时间比当前时间晚的多，相当于查询最新offset；如果输入时间比当前时间早的多，相当于查询最老offset；如果输入空，默认0时间，等价于查询最老offset。 示例值：2023-05-15 16:31:33

3. 输出参数

参数名称	类型	描述
Items	Array of OffsetTimeMap	时间与Offset的对应
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 根据时间查询kafka中的checkpoint消息

查询KafkaTopic中指定时间前最近的checkpoint及其offset

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeOffsetByTime
<公共请求参数>
```

```
{
"SubscribeId": "subs-9jyki7hniw",
"Time": "2023-05-15 16:31:30"
}
```

输出示例

```
{
"Response": {
"Items": [
{
"Offset": 62856,
"PartitionNo": 0
},
{
"Offset": 62842,
"PartitionNo": 1
},
{
"Offset": 62842,
"PartitionNo": 2
},
{
"Offset": 62842,
"PartitionNo": 3
}
],
"RequestId": "087e2780-f3bc-11ed-8dff-ff5f3734b02b"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

重置kafka offset

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ResetConsumerGroupOffset)用于重置订阅消费组的offset。调用DescribeConsumerGroups接口查询消费组状态，只有消费组状态为 Dead 或 Empty 才可以执行重置该操作。否则重置不会生效，接口也不会报错。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResetConsumerGroupOffset。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例id 示例值：subs-7l2ly86kao
TopicName	是	String	订阅的kafka topic 示例值：topic-subs-7l2ly86kao-cdb-5jr9ms95
ConsumerGroupName	是	String	消费组名称。实际的消费组全称形如：consumer-grp-#{SubscribeId}-#{ConsumerGroupName} 示例值：consumer-grp-subs-7l2ly86kao-group-1
PartitionNos.N	是	Array of Integer	需要修改offset的分区编号 示例值：[0,1,2,3]
ResetMode	是	String	重置方式。枚举值为 earliest-从最开始位置开始消费；latest-从最新位置开始消费；datetime-从指定时间前最近的checkpoint开始消费 示例值：earliest
ResetDatetime	否	String	当 ResetMode 为 datetime 时，该项需要填，格式为：Y-m-d h:m:s。如果不填，默认用0时间，效果与earliest相同。 示例值：2022-04-01 11:00:00

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 重置消费组offset

将一个订阅任务的kafka offset重置到2023-05-16 00:00:00附近的checkpoint

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ResetConsumerGroupOffset
<公共请求参数>

{
"ConsumerGroupName": "consumer-grp-subs-635ns8r71g-1",
"PartitionNos": [
"0",
"1",
"2",
"3"
],
"ResetMode": "datetime",
"ResetDatetime": "2023-05-16 00:00:00",
"SubscribeId": "subs-635ns8r71g",
"TopicName": "account-subs-635ns8r71g-1"
}
```

输出示例

```
{
"Response": {
"RequestId": "20461920-b18e-11ec-ae1a-cfe224f4f21f"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

删除消费组

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DeleteConsumerGroup)用于删除一个订阅任务的消费组。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DeleteConsumerGroup。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的 ID 示例值：subs-9jyki7hniw
ConsumerGroupName	是	String	消费组名称。实际的消费组全称形如：consumer-grp-#{SubscribeId}-#{ConsumerGroupName}。请务必保证消费组名称正确。 示例值：consumer-grp-subs-9jyki7hniw-test
AccountName	是	String	账号名称。实际的账户全称形如：account-#{SubscribeId}-#{AccountName}。请务必保证账户名称正确。 示例值：account-subs-9jyki7hniw-test

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除消费组

删除消费组

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DeleteConsumerGroup
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw",
"ConsumerGroupName": "account-subs-9jyki7hniw-test",
```

```
"AccountName": "consumer-grp-subs-9jyki7hniw-test"
}
```

输出示例

```
{
"Response": {
"RequestId": "bef6a6f0-f3b1-11ed-b987-7301347aa36f"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。

错误码	描述
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

重置订阅任务

最近更新时间：2025-04-25 01:29:18

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ResetSubscribe)用于重置订阅实例，重置后，可以重新配置订阅任务。
可以调用 DescribeSubscribeDetail 查询订阅信息判断是否置成功。当SubsStatus变为notStarted时，表示重置成功。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResetSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-9jyki7hniw

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 重置数据订阅任务

重置数据订阅任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=ResetSubscribe
&SubscribeId=subs-ieuwi83j2e
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

修改订阅实例自动续费标识

最近更新时间：2025-04-25 01:29:19

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改订阅实例自动续费标识。只有包年包月的任务修改才有意义，按量计费任务修改后无影响。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeAutoRenewFlag。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID 示例值：subs-9jyki7hniw
AutoRenewFlag	是	Integer	自动续费标识。1-自动续费，0-不自动续费 示例值：1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅自动续费标识

将一个包年包月的订阅任务从不自动续费改为自动续费

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifySubscribeAutoRenewFlag
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw",
"AutoRenewFlag": 1
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

修改数据订阅实例的名称

最近更新时间：2025-04-25 01:29:19

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeName)用于修改数据订阅实例的名称

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeName。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-9jyki7hniw
SubscribeName	是	String	修改后的数据订阅实例的名称，长度限制为[1,60] 示例值：binlog订阅

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅实例的名称

修改数据订阅实例的名称

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifySubscribeName
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw",
"SubscribeName": "subscribe-test"
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

查询订阅实例是否可以退换

最近更新时间：2025-04-25 01:29:20

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeSubscribeReturnable)用于查询订阅任务是否可以销毁和退货。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribeReturnable。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-9jyki7hniw

3. 输出参数

参数名称	类型	描述
IsReturnable	Boolean	实例是否支持退还/退货 示例值：false
ReturnFailMessage	String	不支持退还的原因 示例值：实例已经隔离或者下线，不能退还
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询订阅实例是否可以退换

查询订阅实例是否可以退换

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DescribeSubscribeReturnable
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw"
}
```

输出示例

```
{
"Response": {
"IsReturnable": true,
"ReturnFailMessage": "",
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。

错误码	描述
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

版权所有：腾讯云计算（北京）有限责任公司

销毁订阅任务

最近更新时间：2025-04-25 01:29:19

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（IsolateSubscribe）用于隔离订阅任务。调用后，订阅任务将不能使用。按量计费的任务会停止计费，包年包月的任务会自动退费

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：IsolateSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID 示例值：subs-9jyki7hniw

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 隔离订阅任务

隔离订阅任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: IsolateSubscribe
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw"
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
```

```
  }
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。

错误码	描述
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

下线已隔离的数据订阅任务

最近更新时间：2025-04-25 01:29:19

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（DestroyIsolatedSubscribe）用于下线已隔离的数据订阅实例

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DestroyIsolatedSubscribe。
Version	是	String	公共参数，本接口取值：2021-12-06。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-9jyki7hniw

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 下线已隔离的数据订阅实例

订阅任务已经隔离，立即发起下线任务

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: DestroyIsolatedSubscribe
<公共请求参数>

{
"SubscribeId": "subs-9jyki7hniw"
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
FailedOperation	操作失败。
FailedOperation.FailedOperationError	操作失败。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue	参数取值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation	操作不支持。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

数据结构

最近更新时间：2025-06-24 01:22:37

AdvancedObjectsItem

数据库不一致的详情，mongodb业务用到

被如下接口引用：DescribeCompareReport。

名称	类型	描述
ObjectType	String	对象类型,可能得值有：account,index,shardkey,schema 示例值：account
SrcChunk	String	源端分块 示例值：other.test
DstChunk	String	目标端分块 示例值：other.test
SrcItem	String	源端值 示例值：
DstItem	String	目标端值 示例值：

CheckStep

检查步骤

被如下接口引用：DescribeMigrationCheckJob。

名称	类型	描述
StepNo	Integer	步骤编号 示例值：1
StepId	String	步骤Id， 如：ConnectDBCheck、VersionCheck、SrcPrivilegeCheck等，具体校验项和源目标实例相关 示例值：SrcPrivilegeCheck
StepName	String	步骤名称 示例值：源实例权限检查
StepStatus	String	此检查步骤的结果，pass(校验通过)、failed(校验失败)、notStarted(校验还未开始进行)、blocked(检验阻塞)、warning(校验有告警，但仍通过) 示例值：pass
StepMessage	String	此检查步骤的错误消息 示例值：修改目标库session级别explicit_defaults_for_timestamp参数为ON失败
DetailCheckItems	Array of DetailCheckItem	每个检查步骤里的具体检查项
HasSkipped	Boolean	是否已跳过 示例值：false

CheckStepInfo

校验任务运行详情

被如下接口引用：DescribeMigrationDetail。

名称	类型	描述
StartAt	String	任务开始时间 示例值：2022-07-11 17:16:50

名称	类型	描述
EndAt	String	任务结束时间 示例值：2022-07-11 17:16:50
Progress	ProcessProgress	任务步骤信息

Column

数据同步中的列信息

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
ColumnName	String	否	列名 示例值：c1
NewColumnName	String	否	新列名 示例值：c1_new

CompareAbstractInfo

一致性校验摘要信息

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Options	CompareOptions	校验配置参数
Objects	CompareObject	一致性校验对比对象
Conclusion	String	对比结论: same,different 示例值：same
Status	String	任务状态: success,failed 示例值：success
TotalTables	Integer	总的表数量 示例值：100
CheckedTables	Integer	已校验的表数量 示例值：100
DifferentTables	Integer	不一致的表数量 示例值：1
SkippedTables	Integer	跳过校验的表数量 示例值：1
NearlyTableCount	Integer	预估表总数 示例值：100
DifferentRows	Integer	不一致的数据行数量 示例值：1
SrcSampleRows	Integer	源库行数，当对比类型为**行数对比**时此项有意义 示例值：100000
DstSampleRows	Integer	目标库行数，当对比类型为**行数对比**时此项有意义 示例值：100000
StartedAt	String	开始时间 示例值：2023-12-19 15:54:18
FinishedAt	String	结束时间 示例值：2023-12-19 15:58:18

CompareColumnItem

列选项

被如下接口引用：CreateCompareTask, DescribeCompareReport, ModifyCompareTask。

名称	类型	必选	描述
ColumnName	String	否	列名 示例值：c1

CompareDetailInfo

一致性校验详细信息

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Difference	DifferenceDetail	数据不一致的表详情
Skipped	SkippedDetail	跳过校验的表详情
DifferenceAdvancedObjects	DifferenceAdvancedObjectsDetail	数据库不一致的详情，mongodb业务用到
DifferenceData	DifferenceDataDetail	数据不一致的详情，mongodb业务用到
DifferenceRow	DifferenceRowDetail	数据行不一致的详情，mongodb业务用到

CompareObject

一致性对比对象配置

被如下接口引用：CreateCompareTask, DescribeCompareReport, DescribeCompareTasks, ModifyCompareTask。

名称	类型	必选	描述
ObjectMode	String	是	对象模式 整实例-all,部分对象-partial 示例值：all
ObjectItems	Array of CompareObjectItem	否	对象列表
AdvancedObjects	Array of String	否	高级对象类型，目前只支持mongodb链路。如index(索引),shardkey(片键),schema(库表) 注意：此字段可能返回 null，表示取不到有效值。 示例值：index

CompareObjectItem

一致性校验库表对象

被如下接口引用：CreateCompareTask, DescribeCompareReport, ModifyCompareTask。

名称	类型	必选	描述
DbName	String	是	数据库名 示例值：db1
DbMode	String	是	数据库选择模式: all 为当前对象下的所有对象,partial 为部分对象 示例值：all
SchemaName	String	否	schema名称 示例值：schema1
TableMode	String	否	表选择模式: all 为当前对象下的所有表对象,partial 为部分表对象 示例值：all
Tables	Array of	否	用于一致性校验的表配置，当 TableMode 为 partial 时，需要填写

名称	类型	必选	描述
	CompareTableItem		
ViewMode	String	否	视图选择模式: all 为当前对象下的所有视图对象,partial 为部分视图对象(一致性校验不校验视图，当前参数未启作用) 示例值：all
Views	Array of CompareViewItem	否	用于一致性校验的视图配置，当 ViewMode 为 partial 时， 需要填写(一致性校验不校验视图，当前参数未启作用) 注意：此字段可能返回 null，表示取不到有效值。

CompareOptions

一致性校验选项

被如下接口引用：CreateCompareTask, DescribeCompareReport, DescribeCompareTasks, ModifyCompareTask。

名称	类型	必选	描述
Method	String	否	对比方式：dataCheck(完整数据对比)、sampleDataCheck(抽样数据对比)、rowsCount(行数对比), 默认为dataCheck 示例值：dataCheck
SampleRate	Integer	否	抽样比例;范围0,100。默认为100 示例值：100
ThreadCount	Integer	否	线程数，取值1-8，默认为1 示例值：1

CompareTableItem

用于一致性校验的表配置

被如下接口引用：CreateCompareTask, DescribeCompareReport, ModifyCompareTask。

名称	类型	必选	描述
TableName	String	否	表名称 示例值：t1
ColumnMode	String	否	column 模式，all 为全部，partial 表示部分(该参数仅对数据同步任务有效) 示例值：all
Columns	Array of CompareColumnItem	否	当 ColumnMode 为 partial 时必填(该参数仅对数据同步任务有效)

CompareTaskInfo

数据一致性校验结果

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
CompareTaskId	String	一致性校验任务Id 示例值：dts-amm1jw5q-cmp-bmuum7jk
Status	String	一致性校验结果，包括：unstart(未启动)、running(校验中)、canceled(已终止)、failed(校验任务失败)、inconsistent(不一致)、consistent(一致)、notexist(不存在校验任务) 示例值：consistent

CompareTaskItem

一致性校验对象信息

被如下接口引用：DescribeCompareTasks。

名称	类型	描述
JobId	String	任务id 示例值：dts-e7ukka6g
CompareTaskId	String	对比任务 Id 示例值：dts-e7ukka6g-cmp-c7tnusmo
TaskName	String	对比任务名称 示例值：compare
Status	String	对比任务状态, 可能的值：created - 创建完成；readyRun - 等待运行；running - 运行中；success - 成功；stopping - 结束中；failed - 失败；canceled - 已终止 示例值：success
Config	CompareObject	对比任务配置
CheckProcess	ProcessProgress	对比任务校验详情
CompareProcess	ProcessProgress	对比任务运行详情
Conclusion	String	对比结果, 可能的值：same - 一致；different - 不一致；skipAll - 跳过 示例值：same
CreatedAt	String	任务创建时间 示例值：2022-07-11 17:26:40
StartedAt	String	任务启动时间 示例值：2022-07-11 17:26:40
FinishedAt	String	对比结束时间 示例值：2022-07-11 17:26:40
Method	String	对比类型，dataCheck(完整数据对比)、sampleDataCheck(抽样数据对比)、rowsCount(行数对比) 示例值：dataCheck
Options	CompareOptions	对比配置信息
Message	String	一致性校验提示信息 示例值：success

CompareViewItem

用于一致性校验的视图配置

被如下接口引用：CreateCompareTask, DescribeCompareReport, ModifyCompareTask。

名称	类型	必选	描述
ViewName	String	否	视图名 示例值：view1

ConflictHandleOption

冲突处理里的详细描述

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
ConditionColumn	String	否	条件覆盖的列 示例值：c1
ConditionOperator	String	否	条件覆盖操作 示例值：>
ConditionOrderInSrcAndDst	String	否	条件覆盖优先级处理 示例值：>

ConsistencyOption

数据一致性校验选项， 默认为不开启一致性校验

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
Mode	String	否	一致性检测类型: full(全量检测迁移对象)、noCheck(不检测)、notConfigured(未配置) 示例值：noCheck

DBEndpointInfo

实例信息

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs, ModifyMigrationJob。

名称	类型	必选	描述
Region	String	是	实例所在地域 示例值：ap-guangzhou
AccessType	String	是	实例网络接入类型，如：extranet(外网)、ipv6(公网ipv6)、cvm(云主机自建)、dcg(专线接入)、vpncloud(vpn接入的实例)、cdb(云数据库)、ccn(云联网)、intranet(自研上云)、vpc(私有网络)等，注意具体可选值依赖当前链路 示例值：cdb
DatabaseType	String	是	实例数据库类型，如：mysql,redis,mongodb,postgresql,mariadb,percona 等 示例值：mysql
NodeType	String	是	节点类型，simple表示普通节点、cluster表示集群节点； 对于mongo业务，取值为replicaset(mongodb副本集)、standalone(mongodb单节点)、cluster(mongodb集群)； 对于redis实例，simple(单节点)、cluster-cache(直连集群)、cluster-proxy(代理集群)； 示例值：simple
Info	Array of DBInfo	是	实例具体的连接信息，如ip、port、接入方式等
Supplier	String	否	实例服务提供商，如:"aliyun","others" 示例值：others
ExtraAttr	Array of KeyValuePairOption	否	此参数为数组类型，可以传多个键值对结构对象。 MongoDB可定义如下的参数： 'AuthDatabase':'admin', 'AuthFlag': "1", 'AuthMechanism':"SCRAM-SHA-1", "fetchMethod":"oplog", "connectMode":"srv", "EncryptedConnProtocol":"mongo_atlas_ssl"； 其中fetchMethod表示迁移方式，还可支持change_stream；EncryptedConnProtocol值为mongo_atlas_ssl表示使用atlas ssl连接方式。
DatabaseNetEnv	String	否	数据库所属网络环境，AccessType为云联网(ccn)时必填， UserIDC表示用户IDC、TencentVPC表示腾讯云VPC； 示例值：TencentVPC
ConnectType	String	否	tdsql连接方式：proxy-通过tdsql proxy主机访问各个set节点，注意只有在自研上云的网络环境下才能通过这种方式连接，Info中只需要提供proxy主机信息。set-直连set节点，如选择直连set方式，Info中需要正确填写proxy主机信息及所有set节点信息。源端是tdsqlmysql类型必填。 示例值：proxy
CcnOwnerUin	String	否	云联网网关所属账号，如果云联网网关为其他账号资源需要填写 示例值：212221111

DBInfo

数据库信息

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs, ModifyMigrationJob。

名称	类型	必选	描述
Role	String	否	表示节点角色，针对分布式数据库，如mongodb中的mongos节点。tdsqlmysql的可选项：proxy表示节点类型为主机，set表示节点类型为节点。proxy类型必须填在数组第一项。tdsqlmysql类型的源/目标配置必填。 示例值：mongos
DbKernel	String	否	内核版本，针对mariadb的不同内核版本等 示例值：mysql
Host	String	否	实例的IP地址，对于公网、专线、VPN、云联网、自研上云、VPC等接入方式此项必填 示例值：192.2.3.4
Port	Integer	否	实例的端口，对于公网、云主机自建、专线、VPN、云联网、自研上云、VPC等接入方式此项必填 示例值：3306
User	String	否	实例的用户名 示例值：root
Password	String	否	实例的密码 示例值：password
CvmInstanceId	String	否	CVM实例短ID，格式如：ins-olgl39y8；与云服务器控制台页面显示的实例ID相同；如果接入类型为云主机自建的方式，此项必填 示例值：ins-olgl39y8
UniqVpnGwId	String	否	VPN网关ID，格式如：vpngw-9ghexg7q；如果接入类型为vpncloud的方式，此项必填 示例值：vpngw-9ghexg7q
UniqDcgId	String	否	专线网关ID，格式如：dcg-0rxtqqxb；如果接入类型为专线接入的方式，此项必填 示例值：dcg-0rxtqqxb
InstanceId	String	否	数据库实例ID，格式如：cdb-powiqx8q；如果接入类型为云数据库的方式，此项必填 示例值：cdb-powiqx8q
CcnGwId	String	否	云联网ID，如：ccn-afp6kltc 注意：此字段可能返回 null，表示取不到有效值。 示例值：ccn-afp6kltc
VpcId	String	否	私有网络ID，格式如：vpc-92jblxto；如果接入类型为vpc、vpncloud、ccn、dcg的方式，此项必填 示例值：vpc-92jblxto
SubnetId	String	否	私有网络下的子网ID，格式如：subnet-3paxmkdz；如果接入类型为vpc、vpncloud、ccn、dcg的方式，此项必填 示例值：subnet-3paxmkdz
EngineVersion	String	否	数据库版本，当实例为RDS实例时才有效，格式如：5.6或者5.7，默认为5.6 示例值：5.6
Account	String	否	实例所属账号 示例值：700000123456
AccountRole	String	否	跨账号迁移时的角色,只允许[a-zA-Z0-9-_]+ 示例值：role
AccountMode	String	否	资源所属账号 为空或self(表示本账号内资源)、other(表示其他账户资源) 示例值：self
TmpSecretId	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：AKIDXXXXXX
TmpSecretKey	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：2qZDsXXXXX
TmpToken	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号迁移文档(https://cloud.tencent.com/document/product/571/54117)第4节中关于角色的定义。 示例值：NRUkuXXXXX
EncryptConn	String	否	是否走加密传输、UnEncrypted表示不走加密传输，Encrypted表示走加密传输，默认UnEncrypted 示例值：UnEncrypted

名称	类型	必选	描述
SetId	String	否	tdsql的分片id。如节点类型为set必填。 示例值：set_1693279393_1

DBItem

迁移对象信息，在配置库表视图等对象信息时大小写敏感

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
DbName	String	否	需要迁移或同步的库名，当ObjectMode为partial时，此项必填 示例值：db1
NewDbName	String	否	迁移或同步后的库名，默认与源库相同 示例值：db1_new
SchemaName	String	否	迁移或同步的 schema 示例值：schema1
NewSchemaName	String	否	迁移或同步后的 schema name 示例值：schema1_new
DBMode	String	否	DB选择模式: all(为当前对象下的所有对象)，partial(部分对象)，当ObjectMode为partial时，此项必填 示例值：all
SchemaMode	String	否	schema选择模式: all(为当前对象下的所有对象)，partial(部分对象) 示例值：all
TableMode	String	否	表选择模式: all(为当前对象下的所有对象)，partial(部分对象)，当DBMode为partial时此项必填 示例值：all
Tables	Array of TableItem	否	表图对象集合，当 TableMode 为 partial 时，此项需要填写
ViewMode	String	否	视图选择模式: all 为当前对象下的所有视图对象,partial 为部分视图对象 示例值：all
Views	Array of ViewItem	否	视图对象集合，当 ViewMode 为 partial 时， 此项需要填写
RoleMode	String	否	postgresql独有参数，角色选择模式: all 为当前对象下的所有角色对象,partial 为部分角色对象 示例值：all
Roles	Array of RoleItem	否	postgresql独有参数，当 RoleMode 为 partial 时， 此项需要填写
FunctionMode	String	否	选择要同步的模式，partial为部分，all为整选 示例值：all
TriggerMode	String	否	选择要同步的模式，partial为部分，all为整选 示例值：all
EventMode	String	否	选择要同步的模式，partial为部分，all为整选 示例值：all
ProcedureMode	String	否	选择要同步的模式，partial为部分，all为整选 示例值：all
Functions	Array of String	否	FunctionMode取值为partial时需要填写 示例值：["function_name"]
Procedures	Array of String	否	ProcedureMode取值为partial时需要填写 示例值：["procedure_name"]
Events	Array of String	否	EventMode取值为partial时需要填写 示例值：["event_name"]
Triggers	Array of String	否	TriggerMode取值为partial时需要填写 示例值：["trigger_name"]

Database

需要同步的库表对象

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
DbName	String	否	需要迁移或同步的库名，当ObjectMode为Partial时，此项必填 示例值：db1
NewDbName	String	否	迁移或同步后的库名，默认与源库相同 示例值：db1_new
DbMode	String	否	DB选择模式: All(为当前对象下的所有对象)，Partial(部分对象)，当Mode为Partial时，此项必填。注意，高级对象的同步不依赖此值，如果整库同步此处应该为All。 示例值：All
SchemaName	String	否	迁移或同步的 schema 示例值：sch1
NewSchemaName	String	否	迁移或同步后的 schema name 示例值：sch1_new
TableMode	String	否	表选择模式: All(为当前对象下的所有对象)，Partial(部分对象)，当DBMode为Partial时此项必填，如果整库同步此处应该为All。 示例值：All
Tables	Array of Table	否	表图对象集合，当 TableMode 为 Partial 时，此项需要填写
ViewMode	String	否	视图选择模式: All 为当前对象下的所有视图对象,Partial 为部分视图对象，如果整库同步此处应该为All。 示例值：All
Views	Array of View	否	视图对象集合，当 ViewMode 为 Partial 时， 此项需要填写
FunctionMode	String	否	选择要同步的模式，Partial为部分，All为整选，如果整库同步此处应该为All。 示例值：All
Functions	Array of String	否	FunctionMode取值为Partial时需要填写 示例值：["func_1"]
ProcedureMode	String	否	选择要同步的模式，Partial为部分，All为整选，如果整库同步此处应该为All。 示例值：All
Procedures	Array of String	否	ProcedureMode取值为Partial时需要填写 示例值：["proc_1"]
TriggerMode	String	否	触发器迁移模式，All(为当前对象下的所有对象)，Partial(部分对象)，如果整库同步此处应该为All。数据同步暂不支持此高级对象。 示例值：Partial
Triggers	Array of String	否	当TriggerMode为partial，指定要迁移的触发器名称 示例值：["trigger_1"]
EventMode	String	否	事件迁移模式，All(为当前对象下的所有对象)，Partial(部分对象)，如果整库同步此处应该为All。数据同步暂不支持此高级对象。 示例值：Partial
Events	Array of String	否	当EventMode为partial，指定要迁移的事件名称 示例值：["event_1"]

DatabaseTableObject

迁移对象选项，需要告知迁移服务迁移哪些库表对象

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
ObjectMode	String	是	迁移对象类型 all(全实例)，partial(部分对象) 示例值：partial
Databases	Array of DBItem	否	迁移对象，当 ObjectMode 为 partial 时，不为空
AdvancedObjects	Array of String	否	高级对象类型，如trigger、function、procedure、event。注意：如果要迁移同步高级对象，此配置中应该包含对应的高级对象类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：["trigger"]

DdlOption

数据同步中的ddl同步处理

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
DdlObject	String	否	ddl类型，如Database,Table,View,Index等 示例值：Database
DdlValue	Array of String	否	ddl具体值，对于Database可取值[Create,Drop,Alter] 对于Table可取值[Create,Drop,Alter,Truncate,Rename] 对于View可取值[Create,Drop] 对于Index可取值[Create,Drop] 注意：此字段可能返回 null，表示取不到有效值。 示例值：Alter

DetailCheckItem

每个检查步骤里的具体检查项

被如下接口引用：DescribeMigrationCheckJob。

名称	类型	描述
CheckItemName	String	检查项的名称，如：源实例权限检查 示例值：周边检查
Description	String	检查项详细内容 示例值：周边检查
CheckResult	String	pass(通过)，failed(失败), warning(校验有警告，但仍通过) 示例值：failed
FailureReason	String	检查项失败原因 示例值：如果目标库explicit_defaults_for_timestamp参数不为ON，timestamp列的隐式转换规则可能会导致源库和目标库结构不一致
Solution	String	解决方案 示例值：尝试以下两种方式任意一种即可：1.xxx,2xxx
ErrorLog	Array of String	运行报错日志 示例值：err: Error 1227: Access denied; you need (at least one of) the SUPER privilege(s) for this operation
HelpDoc	Array of String	详细帮助的文档链接 示例值：周边检查: cloud.tencent.com/document/product/571/58690
SkipInfo	String	跳过风险文案 示例值：如果目标库explicit_defaults_for_timestamp参数不为ON，timestamp列的隐式转换规则可能会导致源库和目标库结构不一致

DifferenceAdvancedObjectsDetail

数据库不一致的详情，mongodb业务用到

被如下接口引用：DescribeCompareReport。

名称	类型	描述
TotalCount	Integer	总数 示例值：20
Items	Array of AdvancedObjectsItem	不一致详情 注意：此字段可能返回 null，表示取不到有效值。

DifferenceData

数据不一致详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Db	String	数据库名 示例值：user_c
Table	String	集合 示例值：user_s
SrcChunk	String	源端ID 示例值：ObjectID("626cb354184dd55f78b95f7f")
DstChunk	String	目标端ID 示例值：ObjectID("626cb354184dd55f78b95f7f")
SrcItem	String	源端值 示例值：{"$numberInt":"17"}
DstItem	String	目标端值 示例值：{"$numberInt":"18"}
UpdatedAt	String	更新时间 示例值：2023-10-26 18:54:40

DifferenceDataDetail

mongodb数据不一致性详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
TotalCount	Integer	总数 示例值：10
Items	Array of DifferenceData	mongo数据不一致详细列表 注意：此字段可能返回 null，表示取不到有效值。

DifferenceDetail

数据不一致的表详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
TotalCount	Integer	数据不一致的表数量 示例值：1
Items	Array of DifferenceItem	校验不一致的表详情 注意：此字段可能返回 null，表示取不到有效值。

DifferenceItem

校验不一致的表详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Db	String	数据库名 示例值：db1
Table	String	表名 示例值：t1
Chunk	Integer	分块号 示例值：10
SrcItem	String	源库数值 示例值：1
DstItem	String	目标库数值 示例值：0
IndexName	String	索引名称 示例值：PRIMARY
LowerBoundary	String	索引下边界 示例值：["1","1","2023-12-26"]
UpperBoundary	String	索引上边界 示例值：["+inf"]
CostTime	Float	对比消耗时间,单位为 ms 示例值：0.48
FinishedAt	String	完成时间 示例值：2023-12-26 05:14:24

DifferenceRowDetail

mongodb行数校验不一致性详情结果

被如下接口引用：DescribeCompareReport。

名称	类型	描述
TotalCount	Integer	不一致总数 示例值：100
Items	Array of RowsCountDifference	不一致列表 注意：此字段可能返回 null，表示取不到有效值。

DistributeRule

订阅任务的kafka分区规则。符合库名和表名正则表达式的数据将按照RuleType计算该条数据将被投递的kafka分区。如果配置了多个规则，将按照配置的顺序，第一条命中的规则生效。

被如下接口引用：ConfigureSubscribeJob, DescribeSubscribeDetail, ModifySubscribeObjects。

名称	类型	必选	描述
RuleType	String	是	规则类型。非mongo产品的枚举值为: table-按表名分区，pk-按表名+主键分区，cols-按列名分区。mongo的枚举值为：collection-按集合名分区、collectionAndObjectId-按集合+主键分区。 示例值：table
DbPattern	String	是	库名匹配规则，请填写正则表达式 示例值：db_*
TablePattern	String	是	表名匹配规则，如果 DatabaseType 为 mongodb，则匹配集合名 示例值：tabl_*

名称	类型	必选	描述
Columns	Array of String	否	列名。如果 RuleType 为 cols，此项必填。订阅任务会用该列的值计算分区。mongo没有按列分区，因此也不用传这个字段。 示例值：["id"]

DynamicOptions

数据同步中的选项

被如下接口引用：ModifySyncJobConfig。

名称	类型	必选	描述
OpTypes	Array of String	是	所要同步的DML和DDL的选项，Insert(插入操作)、Update(更新操作)、Delete(删除操作)、DDL(结构同步)，PartialDDL(自定义,和DdlOptions一起起作用)；必填、dts会用该值覆盖原有的值 示例值：["Insert"]
DdlOptions	Array of DdlOption	是	DDL同步选项，具体描述要同步哪些DDL; 当OpTypes取值PartialDDL时、字段不能为空；必填、dts会用该值覆盖原有的值
ConflictHandleType	String	否	冲突处理选项，ReportError(报错)、Ignore(忽略)、Cover(覆盖)、ConditionCover(条件覆盖); 目前目标端为kafka的链路不支持修改该配置 示例值：ReportError
ConflictHandleOption	ConflictHandleOption	否	冲突处理的详细选项，如条件覆盖中的条件行和条件操作；不能部分更新该选项的内部字段；有更新时、需要全量更新该字段
KafkaOption	KafkaOption	否	同步到kafka链路的kafka配置
FilterBeginCommit	Boolean	否	同步到kafka链路是否过滤掉begin和commit消息。目前仅mysql2kafka链路支持 示例值：false
FilterCheckpoint	Boolean	否	同步到kafka链路是否过滤掉checkpoint消息。目前仅mysql2kafka链路支持 示例值：false
DealOfExistSameTable	String	否	同名表的处理，ReportErrorAfterCheck(前置校验并报错，默认)、ExecuteAfterIgnore(忽略并继续执行) 示例值：ReportErrorAfterCheck
StartPosition	String	否	仅增量任务重新设置指定位点 示例值：2025-03-10 11:11:11

Endpoint

数据同步中的描述源端和目的端的信息

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs。

名称	类型	必选	描述
Region	String	否	地域英文名，如：ap-guangzhou 示例值：ap-guangzhou
Role	String	否	节点类型，proxy表示节点类型为主机，set表示节点类型为节点。proxy类型必须填在数组第一项。tdsqlmysql类型的源/目标配置必填 示例值：proxy
DbKernel	String	否	数据库内核类型，tdsql中用于区分不同内核：percona,mariadb,mysql。注意TDSQL-C MySQL、TDSQL PostgreSQL无需填写此项值。 示例值：mariadb
InstanceId	String	否	数据库实例ID，格式如：cdb-powiqx8q 示例值：cdb-wew23ed
Ip	String	否	实例的IP地址，接入类型为非cdb时此项必填 示例值：192.168.1.12

名称	类型	必选	描述
Port	Integer	否	实例端口，接入类型为非cdb时此项必填 示例值：3306
User	String	否	用户名，对于访问需要用户名密码认证的实例必填 示例值：user
Password	String	否	密码，对于访问需要用户名密码认证的实例必填 示例值：password_123
DbName	String	否	数据库名，数据库为cdwpg时，需要提供 示例值：postgress
VpcId	String	否	私有网络ID，对于私有网络、专线、VPN的接入方式此项必填，格式如：vpc-92jblxto 示例值：vpc-23rwfdg
SubnetId	String	否	私有网络下的子网ID，对于私有网络、专线、VPN的接入方式此项必填，格式如：subnet-3paxmkdz 示例值：subnet-tke9sfg
CvmInstanceId	String	否	CVM实例短ID，格式如：ins-olgl39y8，与云服务器控制台页面显示的实例ID相同。如果是CVM自建实例，需要传递此字段 示例值：ins-9emwf2d
UniqDcgId	String	否	专线网关ID，对于专线接入类型此项必填，格式如：dcg-0rxtqqxb 示例值：dcg-9sdfksk
UniqVpnGwId	String	否	VPN网关ID，对于vpn接入类型此项必填，格式如：vpngw-9ghexg7q 示例值：vpngw-3dsfwf2
CcnId	String	否	云联网ID，对于云联网接入类型此项必填，如：ccn-afp6kltc 示例值：ccn-afp6kltc
Supplier	String	否	云厂商类型，当实例为RDS实例时，填写为aliyun, 其他情况均填写others，默认为others 示例值：aliyun
EngineVersion	String	否	数据库版本，当实例为RDS实例时才有效，其他实例忽略，格式如：5.6或者5.7，默认为5.6 示例值：5.6
Account	String	否	实例所属账号，如果为跨账号实例此项必填 示例值：700000123456
AccountMode	String	否	资源所属账号 为空或self(表示本账号内资源)、other(表示跨账号资源) 示例值：other
AccountRole	String	否	跨账号同步时的角色，只允许[a-zA-Z0-9-_]+，如果为跨账号实例此项必填 示例值：dts test_role
RoleExternalId	String	否	外部角色id 示例值：roleId
TmpSecretId	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号同步文档(https://cloud.tencent.com/document/product/571/68729)第4节中关于角色的定义。 示例值：AKID5XXXXX
TmpSecretKey	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号同步文档(https://cloud.tencent.com/document/product/571/68729)第4节中关于角色的定义。 示例值：2qZDXXXXX
TmpToken	String	否	临时密钥Id，可通过申请扮演角色临时访问凭证获取临时密钥 https://cloud.tencent.com/document/product/1312/48197，其中角色资源RoleArn的定义可参考DTS跨账号同步文档(https://cloud.tencent.com/document/product/571/68729)第4节中关于角色的定义。 示例值：JOqqCXXXXX
EncryptConn	String	否	是否走加密传输、UnEncrypted表示不走加密传输，Encrypted表示走加密传输，默认UnEncrypted 示例值：UnEncrypted
DatabaseNetEnv	String	否	数据库所属网络环境，AccessType为云联网(ccn)时必填， UserIDC表示用户IDC、TencentVPC表示腾讯云VPC； 示例值：TencentVPC

名称	类型	必选	描述
CcnOwnerUin	String	否	数据库为跨账号云联网下的实例时、表示云联网所属主账号 示例值：700000123456
ChildInstanceId	String	否	数据库为cynos、且是cynos集群内的一个子数据库实例时、该参数为该子实例的ID 示例值：cynos-ins-xxxx
ChildInstanceType	String	否	数据库为cynos、且是cynos集群内的一个子数据库实例时、该参数为该子实例的类型、例如：只读实例传ro、读写实例传rw 示例值：ro
SetId	String	否	tdsql的分片id。如节点类型为set必填。 示例值：set_1693279393_1(s1)

EndpointItem

数据订阅的实例节点信息

被如下接口引用：ConfigureSubscribeJob, DescribeSubscribeDetail, DescribeSubscribeJobs。

名称	类型	必选	描述
DatabaseRegion	String	是	源库所在地域。如果 AccessType 为 ccn，请填vpc所在地域，因为此时不知道源库在哪个地域。其他接入方式，请填订阅任务所在地域，因为确保订阅任务与源库在同一地域是最优的网络方案。 示例值：ap-guangzhou
User	String	是	用户名 示例值：dts_user
Password	String	是	密码。作为入参时必填，作为出参时为空。 示例值：qJ3e1Wd~
InstanceId	String	否	目标实例ID。如果 AccessType 为 cdb，此项必填。配置InstanceId时会查询并校验实例信息。mysql的查询接口经过了鉴权，请确保子用户有 cdb:DescribeDBInstances 的接口权限。 示例值：cdb-xxxxxx
CvmInstanceId	String	否	云主机ID。如果 AccessType 为 cvm，此项必填。 示例值：ins-xxxxxx
UniqDcgId	String	否	专线网关ID。如果 AccessType 为 dcg，此项必填。 示例值：dcg-xxxxxx
CcnId	String	否	云联网ID。如果 AccessType 为 ccn，此项必填。 示例值：ccn-xxxxxx
UniqVpnGwId	String	否	vpn网关ID。如果 AccessType 为 vpncloud，此项必填。 示例值：vpngw-xxxxxx
VpcId	String	否	VpcID。如果 AccessType 为 dcg\ccn\vpncloud\vpc，此项必填。 示例值：vpc-xxxxxx
SubnetId	String	否	子网ID。如果 AccessType 为 dcg\ccn\vpncloud\vpc，此项必填。 示例值：subnet-xxxxxx
HostName	String	否	数据库地址，支持域名与IP。如果 AccessType 为 dcg\ccn\vpncloud\vpc\extranet\intranet，此项必填。 示例值：9.223.70.88
Port	Integer	否	数据库端口。如果 AccessType 为 dcg\ccn\vpncloud\vpc\extranet\intranet\cvm，此项必填。 示例值：3306
EncryptConn	String	否	是否走加密传输，枚举值：UnEncrypted-不加密，Encrypted-加密。只有mysql支持，不填默认不加密，其他产品不填。 示例值：Encrypted
DatabaseNetEnv	String	否	数据库网络环境。如果 AccessType 为 ccn 此项必填。枚举值：UserIDC-自建idc，TencentVPC-腾讯云，Aws-aws，AliYun-阿里云，Others-其他。

名称	类型	必选	描述
			示例值：TencentVPC
CcnOwnerUin	String	否	云联网网关所属的主账号uin、跨账号云联网需要。 示例值：700000294281
ExtraAttr	Array of KeyValuePairOption	否	为业务添加的额外信息。参数名作key，参数值作value。 tdpg必填参数：PgDatabase-订阅的库名； mongo选填参数：InstanceType-实例类型：replicaset-副本集，cluster-分片集，主要用于控制台跳转到mongo实例页面，如不填不影响任务运行； 全业务选填参数：EngineVersion-内核版本。
ChildInstanceId	String	否	数据库为cynos、且是cynos集群内的一个子数据库实例时、该参数为该子实例的ID 示例值：cynos-ins-xxxx
ChildInstanceType	String	否	数据库为cynos、且是cynos集群内的一个子数据库实例时、该参数为该子实例的类型、例如：只读实例传ro、读写实例传rw 示例值：ro

ErrInfo

错误信息及其解决方案

被如下接口引用：DescribeSyncJobs。

名称	类型	描述
Reason	String	错误原因 示例值：连接源或者目标实例出现网络错误。
Message	String	错误信息 示例值：dial tcp : i/o timeout
Solution	String	解决方案 示例值：请检查是否存在以上情况并解决。

ErrorInfoItem

任务错误信息

被如下接口引用：DescribeMigrationDetail。

名称	类型	描述
Code	String	错误码 示例值：ResourceNotFound
Solution	String	解决方案 示例值：请确认库表存在后重试
ErrorLog	String	错误日志信息 示例值：execute fail: Error 1146: Table 'tencentdb.eg_dts-4lcse9uh' doesn't exist
HelpDoc	String	文档提示 示例值：请确认库表存在后重试

GroupInfo

kafka消费者组详情

被如下接口引用：DescribeConsumerGroups。

名称	类型	描述
Account	String	消费者组账号 示例值：account-subs-47damshnra-group-1

名称	类型	描述
ConsumerGroupName	String	消费者组名称 示例值：consumer-grp-subs-47damshnra-jason
Description	String	消费者组备注 示例值：binlog订阅
ConsumerGroupOffset	Integer	消费组偏移量。该字段是为了兼容以前单Partition的情况，取值为最后一个分区的偏移量。各分区的偏移量详见StateOfPartition字段 示例值：-1
ConsumerGroupLag	Integer	消费组未消费的数据量。该字段是为了兼容以前单Partition的情况，取值为最后一个分区未消费的数据量。各分区未消费数据量详见StateOfPartition字段 示例值：0
Latency	Integer	消费延迟(单位为秒) 示例值：9
StateOfPartition	Array of MonitorInfo	各分区的消费状态
CreatedAt	String	消费者组创建时间，格式为YYYY-MM-DD hh:mm:ss 示例值：2022-03-25 17:31:31
UpdatedAt	String	消费者组修改时间，格式为YYYY-MM-DD hh:mm:ss 示例值：2022-03-25 17:31:31
ConsumerGroupState	String	消费者组状态，包括Dead、Empty、Stable等，只有Dead和Empty两种状态可以执行reset操作 示例值：Stable
PartitionAssignment	Array of PartitionAssignment	每个消费者正在消费的分区

JobItem

迁移任务列表

被如下接口引用：DescribeMigrationJobs。

名称	类型	描述
JobId	String	数据迁移任务ID 示例值：dts-amm1jw5q
JobName	String	数据迁移任务名称 示例值：dts-amm1jw5q
CreateTime	String	任务创建(提交)时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
UpdateTime	String	任务更新时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
StartTime	String	任务开始执行时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
EndTime	String	任务执行结束时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2022-07-11 16:20:49
BriefMsg	String	迁移任务错误信息 示例值：execute fail: Error 1146: Table '**tencentdb**.eg_dts-4lcse9uh' doesn't exist
Status	String	任务状态，取值为：creating(创建中)、created(创建完成)、checking(校验中)、checkPass(校验通过)、checkNotPass(校验不通过)、readyRun(准备运行)、running(任务运行)、readyComplete(准备完成)、success(任务成功)、failed(任务失败)、stopping(中止中)、completing(完成中)、 pausing(暂停中)、 manualPaused(已暂停) 示例值：created

名称	类型	描述
RunMode	String	任务运行模式，值包括：immediate(立即运行)，timed(定时运行) 示例值：immediate
ExpectRunTime	String	期待启动时间，当RunMode取值为timed时，此值必填，形如：2022-07-11 16:20:49 示例值：2022-07-11 16:20:49
Action	MigrateAction	任务操作信息
StepInfo	MigrateDetailInfo	迁移执行过程信息
SrcInfo	DBEndpointInfo	源实例信息
DstInfo	DBEndpointInfo	目标端信息
CompareTask	CompareTaskInfo	数据一致性校验结果
TradeInfo	TradeInfo	计费状态信息
Tags	Array of TagItem	标签信息
AutoRetryTimeRangeMinutes	Integer	自动重试时间段信息 示例值：5
DumperResumeCtrl	String	全量导出可重入标识：enum::"yes"/"no"。yes表示当前任务可重入、no表示当前任务处于全量导出且不可重入阶段；如果在该值为no时重启任务导出流程不支持断点续传 示例值：yes

KafkaOption

目标端为kafka时添加的同步选项字段

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
DataType	String	否	投递到kafka的数据类型，如Avro,Json,canal-pb,canal-json 示例值：Avro,Json
TopicType	String	否	同步topic策略，如Single（集中投递到单topic）,Multi (自定义topic名称) 示例值：Single（集中投递到单topic）,Multi (自定义topic名称)
DDLTopicName	String	否	用于存储ddl的topic 示例值：topic-ddl
TopicRules	Array of TopicRule	否	单topic和自定义topic的描述

KeyValuePairOption

存放配置时的额外信息

被如下接口引用：ConfigureSubscribeJob, DescribeMigrationDetail, DescribeMigrationJobs, DescribeSubscribeDetail, ModifyMigrateRuntimeAttribute, ModifyMigrationJob。

名称	类型	必选	描述
Key	String	否	选项key 示例值：AuthMechanism
Value	String	否	选项value 示例值：SCRAM-SHA-1

MigrateAction

任务操作信息，包含迁移任务的所有操作列表，及迁移任务在当前状态下允许的操作列表

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
AllAction	Array of String	任务的所有操作列表 示例值：["view","modify","check","start","configure","stop","complete","
AllowedAction	Array of String	任务在当前状态下允许的操作列表 示例值：["view","modify","configure","view","isolate","resize"]

MigrateDBItem

查询迁移实例列表的实例对象

被如下接口引用：DescribeMigrateDBInstances。

名称	类型	描述
InstanceId	String	实例ID 示例值：cdb-9djwowo2
InstanceName	String	实例名称 示例值：实例名称
Vip	String	实例Vip 示例值：192.168.1.1
Vport	Integer	实例Vport 示例值：3306
Usable	Integer	是否可以作为迁移对象，1-可以，0-不可以 示例值：1
Hint	String	不可以作为迁移对象的原因 示例值：当前实例不可以作为迁移源

MigrateDetailInfo

迁移执行过程信息

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
StepAll	Integer	总步骤数 示例值：3
StepNow	Integer	当前步骤 示例值：3
MasterSlaveDistance	Integer	主从差距，MB；只在任务正常，迁移或者同步的最后一步（追Binlog的阶段才有校），如果是非法值，返回-1 示例值：0
SecondsBehindMaster	Integer	主从差距，秒；只在任务正常，迁移或者同步的最后一步（追Binlog的阶段才有校），如果是非法值，返回-1 示例值：0
StepInfo	Array of StepDetailInfo	步骤信息

MigrateOption

迁移选项，描述任务如何执行迁移等一系列配置信息

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
DatabaseTable	DatabaseTableObject	是	迁移对象选项，需要告知迁移服务迁移哪些库表对象

名称	类型	必选	描述
MigrateType	String	否	迁移类型，full(全量迁移)，structure(结构迁移)，fullAndIncrement(全量加增量迁移)， 默认为fullAndIncrement;注意redis,keewidb产品只支持fullAndIncrement类型。 示例值：fullAndIncrement
Consistency	ConsistencyOption	否	数据一致性校验选项， 默认为不开启一致性校验
IsMigrateAccount	Boolean	否	是否迁移账号，true(迁移账号)，false(不迁移账号) 示例值：false
IsOverrideRoot	Boolean	否	是否用源库Root账户覆盖目标库，值包括：false-不覆盖，true-覆盖，选择库表或者结构迁移时应该为false，注意只对旧版迁移有效 示例值：false
IsDstReadOnly	Boolean	否	是否在迁移时设置目标库只读(仅对mysql有效)，true(设置只读)、false(不设置只读，默认此值) 示例值：false
ExtraAttr	Array of KeyValuePairOption	否	其他附加信息，对于特定库可设置额外参数，Redis可定义如下的参数: ["DstWriteMode":normal, 目标库写入模式,可取值clearData(清空目标实例数据)、overwrite(以覆盖写的方式执行任务)、normal(跟正常流程一样，不做额外动作) "IsDstReadOnly":true, 是否在迁移时设置目标库只读,true(设置只读)、false(不设置只读) "ClientOutputBufferHardLimit":512, 从机缓冲区的硬性容量限制(MB) "ClientOutputBufferSoftLimit":512, 从机缓冲区的软性容量限制(MB) "ClientOutputBufferPersistTime":60, 从机缓冲区的软性限制持续时间(秒) "ReplBacklogSize":512, 环形缓冲区容量限制(MB) "ReplTimeout":120， 复制超时时间(秒) "IsExpireKey":"true",过期key自动淘汰]
MigrateWay	String	否	pgsql迁移分类：logical(逻辑迁移)、physical(物理迁移) 示例值：logical

ModifiedSubscribeObject

数据数据订阅的对象，用于修改订阅对象接口。与SubscribeObject结构类似，只是类型和参数名不同。

被如下接口引用：ModifySubscribeObjects。

名称	类型	必选	描述
ObjectsType	Integer	是	订阅对象的类型，枚举值为：0-库，1-表(该值对于mongo任务来说，是集合）。 注意：mongo只支持全实例、单库或者单集合订阅，因此该字段不要与SubscribeObjectType冲突。如：SubscribeObjectType=4，表示mongo单库订阅，那么该字段应该传0。 示例值：1
DatabaseName	String	是	订阅数据库的名称 示例值：db_1
TableNames	Array of String	否	订阅数据库中表(或集合)的名称。如果 ObjectsType 为 1，那么此字段为必填，且不为空； 示例值：["tabl_1"]

MonitorInfo

kafka消费者组的分区详情

被如下接口引用：DescribeConsumerGroups。

名称	类型	描述
PartitionNo	Integer	当前分区的编号，从0开始 示例值：1
ConsumerGroupOffset	Integer	当前分区的偏移量 示例值：23859978
ConsumerGroupLag	Integer	当前分区未消费的数据量 示例值：0
Latency	Integer	当前分区的消费延迟(单位为秒) 示例值：1

Objects

同步的数据库对对象描述

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
Mode	String	否	同步对象类型 Partial(部分对象) 示例值：Partial
Databases	Array of Database	否	同步对象，当 Mode 为 Partial 时，不为空 注意：此字段可能返回 null，表示取不到有效值。
AdvancedObjects	Array of String	否	高级对象类型，如function、procedure。注意：如果要迁移同步高级对象，此配置中应该包含对应的高级对象类型。当需要同步高级对象时，初始化类型必须包含结构初始化类型，即任务的Options.InitType字段值为Structure或Full 示例值：["function"]
OnlineDDL	OnlineDDL	否	OnlineDDL类型，冗余字段不做配置用途

OffsetTimeMap

数据订阅kafka分区中checkpoint信息

被如下接口引用：DescribeOffsetByTime。

名称	类型	描述
PartitionNo	Integer	kafka分区编号 示例值：0
Offset	Integer	kafka offset 示例值：1234565

OnlineDDL

OnlineDDL类型

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
Status	String	是	状态 示例值：ON

Options

数据同步中的选项

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs。

名称	类型	必选	描述
InitType	String	否	同步初始化选项，Data(全量数据初始化)、Structure(结构初始化)、Full(全量数据且结构初始化，默认)、None(仅增量) 示例值：Full
DealOfExistSameTable	String	否	同名表的处理，ReportErrorAfterCheck(前置校验并报错，默认)、ExecuteAfterIgnore(忽略并继续执行) 示例值：ReportErrorAfterCheck
ConflictHandleType	String	否	冲突处理选项，ReportError(报错，默认为该值)、Ignore(忽略)、Cover(覆盖)、ConditionCover(条件覆盖) 示例值：ReportError
AddAdditionalColumn	Boolean	否	是否添加附加列 示例值：false

名称	类型	必选	描述
OpTypes	Array of String	否	所要同步的DML和DDL的选项，Insert(插入操作)、Update(更新操作)、Delete(删除操作)、DDL(结构同步)， PartialDDL(自定义,和DdlOptions一起配合使用)。注意，这里至少需要包含DML中的一种。 注意：此字段可能返回 null，表示取不到有效值。 示例值：["Insert","Update","Delete","DDL"]
ConflictHandleOption	ConflictHandleOption	否	冲突处理的详细选项，如条件覆盖中的条件行和条件操作 示例值：ConflictHandleOption
DdlOptions	Array of DdlOption	否	DDL同步选项，具体描述要同步哪些DDL 注意：此字段可能返回 null，表示取不到有效值。 示例值：DdlOption
KafkaOption	KafkaOption	否	kafka同步选项 示例值：KafkaOption
RateLimitOption	RateLimitOption	否	任务限速信息 示例值：RateLimitOption
AutoRetryTimeRangeMinutes	Integer	否	自动重试的时间窗口设置 示例值：20
StartPosition	String	否	同步到kafka链路指定位点。目前只支持时间格式：2023-12-20T19:24:23+08:00。如果没有指定位点，为空。 示例值：2023-12-20T19:24:23+08:00
FilterBeginCommit	Boolean	否	同步到kafka链路是否过滤掉begin和commit消息。目前仅mysql2kafka链路支持 示例值：false
FilterCheckpoint	Boolean	否	同步到kafka链路是否过滤掉checkpoint消息。目前仅mysql2kafka链路支持 示例值：false

PartitionAssignment

数据订阅中kafka消费者组的分区分配情况。该数据是实时查询的，如果需要最新数据，需重新掉接口查询。

被如下接口引用：DescribeConsumerGroups。

名称	类型	描述
ClientId	String	消费者的clientId 示例值：ding
PartitionNo	Array of Integer	该消费者正在消费的分区 注意：此字段可能返回 null，表示取不到有效值。 示例值：[0,1]

PipelineInfo

mongo输出聚合设置。输出默认 Change Event

被如下接口引用：ConfigureSubscribeJob, DescribeSubscribeDetail, ModifySubscribeObjects。

名称	类型	必选	描述
AggOp	String	否	聚合运算符：$addFields、$match、$project、$replaceRoot、$redact、$replaceWith、$set、$unset。其中$replaceWith、$set、$unset 只有当订阅实例是4.2及以上版本可选。 示例值：$addFields
AggCmd	String	否	聚合表达式。必须是json格式 示例值：{"a":""}

ProcessProgress

任务步骤信息

被如下接口引用：DescribeCompareTasks, DescribeMigrationDetail。

名称	类型	描述
Status	String	步骤的状态， 包括：notStarted(未开始)、running(运行中)、success(成功)、failed(失败)等 示例值：success
Percent	Integer	进度信息 示例值：0
StepAll	Integer	总的步骤数 示例值：10
StepNow	Integer	当前进行的步骤 示例值：3
Message	String	当前步骤输出提示信息 示例值：ok
Steps	Array of StepDetailInfo	步骤信息

ProcessStepTip

错误信息及告警信息对象

被如下接口引用：DescribeCompareTasks, DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
Message	String	提示信息 示例值：execute fail: Error 1146: Table 'tencentdb.eg_dts-4lcse9uh' doesn't exist
Solution	String	解决方案 示例值：1. 请到目标库执行'show create table xxx'，确认该表是否存在；如果不存在，请在目标库中手动建表，然后重试任务。
HelpDoc	String	文档提示 示例值：库表不存在

RateLimitOption

迁移和同步任务限速的详细信息

被如下接口引用：ConfigureSyncJob, DescribeMigrationDetail, DescribeSyncJobs。

名称	类型	必选	描述
CurrentDumpThread	Integer	是	当前生效的全量导出线程数，配置任务时可调整该字段值，注意：如果不设置或设置为0则表示保持当前值，最大值为16 示例值：8
DefaultDumpThread	Integer	是	默认的全量导出线程数，该字段仅在出参有意义 示例值：8
CurrentDumpRps	Integer	是	当前生效的全量导出Rps，配置任务时可调整该字段值，注意：如果不设置或设置为0则表示保持当前值，最大值为50000000 示例值：400000
DefaultDumpRps	Integer	是	默认的全量导出Rps，该字段仅在出参有意义 示例值：400000
CurrentLoadThread	Integer	是	当前生效的全量导入线程数，配置任务时可调整该字段值，注意：如果不设置或设置为0则表示保持当前值，最大值为16 示例值：8
DefaultLoadThread	Integer	是	默认的全量导入线程数，该字段仅在出参有意义 示例值：8
CurrentLoadRps	Integer	是	当前生效的全量导入Rps，配置任务时可调整该字段值，注意：如果不设置或设置为0则表示保持当前值，最大值为50000000 示例值：400000

名称	类型	必选	描述
DefaultLoadRps	Integer	是	默认的全量导入Rps，该字段仅在出参有意义 示例值：400000
CurrentSinkerThread	Integer	是	当前生效的增量导入线程数，配置任务时可调整该字段值，注意：如果不设置或设置为0则表示保持当前值，最大值为128 示例值：32
DefaultSinkerThread	Integer	是	默认的增量导入线程数，该字段仅在出参有意义 示例值：32
HasUserSetRateLimit	String	是	enum:"no"/"yes"、no表示用户未设置过限速、yes表示设置过限速，该字段仅在出参有意义 示例值：no

RoleItem

角色对象，postgresql独有参数

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
RoleName	String	否	角色名称 示例值：r1
NewRoleName	String	否	迁移后的角色名称 示例值：r1_new

RowsCountDifference

mongodb行校验不一致详细信息

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Db	String	数据库名 示例值：user
Table	String	集合 示例值：user_core
SrcCount	Integer	源端行数 示例值：100
DstCount	Integer	目标端行数 示例值：105

SkippedDetail

跳过校验的表详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
TotalCount	Integer	跳过的表数量 示例值：1
Items	Array of SkippedItem	跳过校验的表详情

SkippedItem

跳过校验的表详情

被如下接口引用：DescribeCompareReport。

名称	类型	描述
Db	String	数据库名 示例值：db1
Table	String	表名 示例值：t1
Reason	String	未发起检查的原因 示例值：engine not support, only support MyISAM or InnoDB

StepDetailInfo

步骤信息

被如下接口引用：DescribeCompareTasks, DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
StepNo	Integer	步骤序列 示例值：1
StepName	String	步骤展现名称 示例值：源库导出
StepId	String	步骤英文标识 示例值：dumper
Status	String	步骤状态:success(成功)、failed(失败)、running(执行中)、notStarted(未执行)、默认为notStarted 示例值：running
StartTime	String	当前步骤开始的时间，格式为"yyyy-mm-dd hh:mm:ss"，该字段不存在或者为空是无意义 注意：此字段可能返回 null，表示取不到有效值。 示例值：2022-07-11 17:22:43
StepMessage	String	步骤错误信息 示例值：i/o timeout
Percent	Integer	执行进度 示例值：0
Errors	Array of ProcessStepTip	错误信息
Warnings	Array of ProcessStepTip	告警提示

StepInfo

单个步骤的详细信息

被如下接口引用：DescribeCheckSyncJobResult, DescribeModifyCheckSyncJobResult, DescribeSyncJobs。

名称	类型	必选	描述
StepNo	Integer	否	步骤编号 示例值：1
StepName	String	否	步骤名 示例值：连接DB检查
StepId	String	否	步骤标号 示例值：ConnectDBCheck
Status	String	否	当前步骤状态,可能返回有 notStarted(未开始)、running(校验中)、failed(校验任务失败)、finished(完成)、skipped(跳过)、paused(暂停) 示例值：finished
StartTime	String	否	步骤开始时间，可能为空

名称	类型	必选	描述
			示例值：2023-12-19 15:54:18
Errors	Array of StepTip	否	错误信息 注意：此字段可能返回 null，表示取不到有效值。
Warnings	Array of StepTip	否	警告信息 注意：此字段可能返回 null，表示取不到有效值。
Progress	Integer	否	当前步骤进度，范围为[0-100]，若为-1表示当前步骤不支持查看进度 示例值：100

StepTip

当前步骤错误信息或者警告信息

被如下接口引用：DescribeCheckSyncJobResult, DescribeModifyCheckSyncJobResult。

名称	类型	必选	描述
Code	String	否	错误码 示例值：Error: OptimizeCheck
Message	String	否	错误信息 示例值：修改目标库session级别explicit_defaults_for_timestamp参数为ON失败，错误信息详情
Solution	String	否	解决方式 示例值：尝试以下解决方案：1.xxx，2.xxx
HelpDoc	String	否	帮助文档 示例值：周边检查: cloud.tencent.com/document/product/571/58690
SkipInfo	String	否	当前步骤跳过信息 示例值：如果目标库explicit_defaults_for_timestamp参数不为ON，timestamp列的隐式转换规则可能会导致源库和目标库结构不一致

SubsErr

订阅报错信息

被如下接口引用：DescribeSubscribeDetail, DescribeSubscribeJobs。

名称	类型	描述
Message	String	报错信息 示例值：kafka: Failed to produce message to topic
Reason	String	报错原因 示例值：kafka 异常 1. 如果是数据订阅链路，可能是内部Kafka组件生产异常，后端服务会自动重试并自愈，请刷新一下页面，关注任务的状态流转 2. 如果是数据同步到kafka链路，可能是用户的kafka异常，导致数据无法写入
Solution	String	建议的修复方案 示例值：1. 如果数据订阅链路超过10分钟后刷新，状态仍然没有流转，请提交工单协助解决。 2. 如果是数据同步到kafka链路，请前往kafka控制台排查

SubscribeCheckStepInfo

订阅校验任务的各步骤信息。

被如下接口引用：DescribeSubscribeCheckJob。

名称	类型	描述
StepName	String	步骤名称 示例值：连接DB检查
StepId	String	步骤Id

名称	类型	描述
		示例值：ConnectDBCheck
StepNo	Integer	步骤编号，从 1 开始 示例值：1
Status	String	当前步骤状态，可能值为 notStarted,running,finished,failed 示例值：running
Percent	Integer	当前步骤进度 示例值：100
Errors	Array of SubscribeCheckStepTip	错误提示
Warnings	Array of SubscribeCheckStepTip	告警提示

SubscribeCheckStepTip

订阅校验任务的提示信息

被如下接口引用：DescribeSubscribeCheckJob。

名称	类型	描述
Message	String	错误或告警的详细信息 示例值：kafka: Failed to produce message to topic
HelpDoc	String	帮助文档 示例值：https://cloud.tencent.com/document/product/571/92710

SubscribeInfo

订阅实例信息

被如下接口引用：DescribeSubscribeJobs。

名称	类型	描述
SubscribeId	String	数据订阅的实例ID 示例值：subs-9jyki7hniw
SubscribeName	String	数据订阅实例的名称 示例值：binlog订阅
Topic	String	订阅实例发送数据的kafka topic 示例值：topic-subs-l4d3a7izai-cdb-24ifcx43
Product	String	订阅实例的类型，目前支持 cynosdbmysql,mariadb,mongodb,mysql,percona,tdpg,tdsqlpercona(tdsqlmysql) 示例值：mysql
InstanceId	String	订阅的数据库实例ID（如果订阅的是云数据库）如果实例不是腾讯云上的，此值为空。 示例值：cdb-kdxona7h
InstanceStatus	String	云数据库状态：running 运行中，isolated 已隔离，offline 已下线。如果不是云上，此值为空 示例值：running
Status	String	数据订阅生命周期状态，可能的值为：正常 normal, 隔离中 isolating, 已隔离 isolated, 下线中 offlining, 按量转包年包月中 post2PrePayIng 示例值：normal
SubsStatus	String	数据订阅状态，可能的值为：未启动 notStarted, 校验中 checking, 校验不通过 checkNotPass, 校验通过 checkPass, 启动中 starting, 运行中 running, 异常出错 error 示例值：running
ModifyTime	String	上次修改时间，时间格式如：Y-m-d h:m:s 示例值：2023-05-15 21:29:05

名称	类型	描述
CreateTime	String	创建时间，时间格式如：Y-m-d h:m:s 示例值：2023-05-15 21:29:05
IsolateTime	String	隔离时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：2023-05-15 21:29:05
ExpireTime	String	包年包月任务的到期时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：2023-05-15 21:29:05
OfflineTime	String	下线时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：2023-05-15 21:29:05
PayType	Integer	计费方式，0 - 包年包月，1 - 按量计费 示例值：0
AutoRenewFlag	Integer	自动续费标识。只有当 PayType=0，该值才有意义。枚举值：0-不自动续费，1-自动续费 示例值：1
Region	String	数据订阅实例所属地域 示例值：ap-guangzhou
AccessType	String	接入方式。枚举值：extranet(公网) vpncloud(vpn接入) dcg(专线接入) ccn(云联网) cdb(云数据库) cvm(云主机自建) intranet(自研上云) vpc(私有网络vpc) 示例值：cdb
Endpoints	Array of EndpointItem	数据库节点信息
SubscribeVersion	String	数据订阅版本, 当前只支持 kafka 版本。 示例值：kafka
Tags	Array of TagItem	标签 注意：此字段可能返回 null，表示取不到有效值。
Errors	Array of SubsErr	任务报错信息，如果有的话。 注意：此字段可能返回 null，表示取不到有效值。

SubscribeKafkaConfig

订阅的kafka分区数和分区规则。mariadb，percona，tdsqlmysql，tdpg不支持自定义分区，所以DistributeRules和DefaultRuleType可以不填，但是NumberOfPartitions是必填。

被如下接口引用：ConfigureSubscribeJob, DescribeSubscribeDetail。

名称	类型	必选	描述
NumberOfPartitions	Integer	是	kafka分区数量，枚举值为1，4，8 示例值：8
DistributeRules	Array of DistributeRule	否	分区规则。当NumberOfPartitions > 1时，该项必填。
DefaultRuleType	String	否	默认分区策略。当NumberOfPartitions > 1时，该项必填。不满足DistributeRules中正则表达式的数据，将按照默认分区策略计算分区。 非mongo产品的枚举值为: table-按表名分区，pk-按表名+主键分区。mongo的枚举值为: collection-按集合名分区。 该字段与DistributeRules搭配使用，如果配置了该字段，视为配置了一条DistributeRules。 示例值：table

SubscribeObject

订阅的数据库表信息，用于配置和查询订阅任务接口。

被如下接口引用：ConfigureSubscribeJob, DescribeSubscribeDetail。

名称	类型	必选	描述
ObjectType	String	是	订阅数据的类型，枚举值：database-数据库，table-数据库的表(如果 DatabaseType 为 mongodb，则表示集合) 示例值：table
Database	String	是	订阅的数据库名称 示例值：db_1
Tables	Array of String	否	订阅数据库中表的名称。如果 DatabaseType 为 mongodb，填集合名。mongodb只支持订阅单库或者单集合。 注意：此字段可能返回 null，表示取不到有效值。 示例值：["tabl_1"]

SyncDBEndpointInfos

数据同步配置多节点数据库的节点信息。多节点数据库，如tdsqlmysql使用该结构；单节点数据库，如mysql使用Endpoint。

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs。

名称	类型	必选	描述
Region	String	是	数据库所在地域 示例值：ap-guangzhou
AccessType	String	是	实例网络接入类型，如：extranet(外网)、ipv6(公网ipv6)、cvm(云主机自建)、dcg(专线接入)、vpncloud(vpn接入的实例)、cdb(云数据库)、ccn(云联网)、intranet(自研上云)、vpc(私有网络)等，注意具体可选值依赖当前链路 示例值：cdb
DatabaseType	String	是	实例数据库类型，如：mysql,redis,mongodb,postgresql,mariadb,percona 等 示例值：tdsqlmysql
Info	Array of Endpoint	是	数据库信息。注意：如果数据库类型为tdsqlmysql，此处Endpoint数组的顺序应满足规则：proxy节点放在set节点之前。如果SrcConnectType选择proxy接入则只需要填写proxy节点即可。如果选择set接入，数组中第一个set节点必须是shardkey范围起始为0的分片 注意：此字段可能返回 null，表示取不到有效值。

SyncDetailInfo

同步任务的步骤信息

被如下接口引用：DescribeSyncJobs。

名称	类型	描述
StepAll	Integer	总步骤数 示例值：4
StepNow	Integer	当前步骤 示例值：4
Progress	Integer	总体进度 示例值：0
CurrentStepProgress	Integer	当前步骤进度，范围为[0-100]，若为-1表示当前步骤不支持查看进度 示例值：50
MasterSlaveDistance	Integer	同步两端数据量差距 示例值：0
SecondsBehindMaster	Integer	同步两端时间差距 示例值：0
Message	String	总体描述信息 示例值：
StepInfos	Array of StepInfo	详细步骤信息 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
CauseOfCompareDisable	String	不能发起一致性校验的原因 示例值：当前任务开启了DML过滤
ErrInfo	ErrInfo	任务的错误和解决方案信息

SyncJobInfo

同步任务信息

被如下接口引用：DescribeSyncJobs。

名称	类型	描述
JobId	String	同步任务id，如：sync-btso140 示例值：sync-btso140
JobName	String	同步任务名 示例值：sync-btso140
PayMode	String	付款方式，PostPay(按量付费)、PrePay(包年包月) 示例值：PostPay
RunMode	String	运行模式，Immediate(表示立即运行，默认为此项值)、Timed(表示定时运行) 示例值：Immediate
ExpectRunTime	String	期待运行时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
AllActions	Array of String	支持的所有操作 示例值： ["pause","stop","limitRate","view","resize","isolate","postPay2prePay","configureC
Actions	Array of String	当前状态能进行的操作 注意：此字段可能返回 null，表示取不到有效值。 示例值： ["pause","stop","limitRate","view","resize","isolate","postPay2prePay","configureC
Options	Options	同步选项
Objects	Objects	同步库表对象
Specification	String	任务规格 示例值：large
ExpireTime	String	过期时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
SrcRegion	String	源端地域，如：ap-guangzhou等 示例值：ap-guangzhou
SrcDatabaseType	String	源端数据库类型，mysql,cynosdbmysql,tdapg,tdpg,tdsqlmysql等 示例值：mysql
SrcAccessType	String	源端接入类型，cdb(云数据库)、cvm(云主机自建)、vpc(私有网络)、extranet(外网)、vpnclo接入)、dcg(专线接入)、ccn(云联网)、intranet(自研上云) 示例值：cdb
SrcInfo	Endpoint	源端信息，单节点数据库使用
SrcNodeType	String	枚举值：cluster、single。源库为单节点数据库使用single，多节点使用cluster 示例值：cluster
SrcInfos	SyncDBEndpointInfos	源端信息，多节点数据库使用
DstRegion	String	目标端地域，如：ap-guangzhou等 示例值：ap-guangzhou
DstDatabaseType	String	目标端数据库类型，mysql,cynosdbmysql,tdapg,tdpg,tdsqlmysql等 示例值：mysql

名称	类型	描述
DstAccessType	String	目标端接入类型，cdb(云数据库)、cvm(云主机自建)、vpc(私有网络)、extranet(外网)、vpncloud(vpn接入)、dcg(专线接入)、ccn(云联网)、intranet(自研上云) 示例值：cdb
DstInfo	Endpoint	目标端信息，单节点数据库使用
DstNodeType	String	枚举值：cluster、single。目标库为单节点数据库使用single，多节点使用cluster 示例值：cluster
DstInfos	SyncDBEndpointInfos	目标端信息，多节点数据库使用
CreateTime	String	创建时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
StartTime	String	开始时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
Status	String	任务状态，UnInitialized(未初始化)、Initialized(已初始化)、Checking(校验中)、CheckPas 通过)、CheckNotPass(校验不通过)、ReadyRunning(准备运行)、Running(运行中)、Paus 停中)、Paused(已暂停)、Stopping(停止中)、Stopped(已结束)、ResumableErr(任务错误) Resuming(恢复中)、Failed(失败)、Released(已释放)、Resetting(重置中)、Unknown(未 示例值：CheckPass
EndTime	String	结束时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
Tags	Array of TagItem	标签相关信息
Detail	SyncDetailInfo	同步任务运行步骤信息
TradeStatus	String	用于计费的状态，可能取值有：Normal(正常状态)、Resizing(变配中)、Renewing(续费中)、 Isolating(隔离中)、Isolated(已隔离)、Offlining(下线中)、Offlined(已下线)、NotBilled(未计 Recovering(解隔离)、PostPay2Prepaying(按量计费转包年包月中)、PrePay2Postpaying 月转按量计费中) 示例值：Normal
InstanceClass	String	同步链路规格，如micro,small,medium,large 示例值：medium
AutoRenew	Integer	自动续费标识，当PayMode值为PrePay则此项配置有意义，取值为：1（表示自动续费）、0（不 费） 示例值：0
OfflineTime	String	下线时间，格式为 yyyy-mm-dd hh:mm:ss 示例值：2023-12-08 16:24:32
OptObjStatus	String	动态修改对象，修改任务的状态等 示例值：Normal
AutoRetryTimeRangeMinutes	Integer	自动重试时间段设置 示例值：5
DumperResumeCtrl	String	全量导出可重入标识：enum::"yes"/"no"。yes表示当前任务可重入、no表示当前任务处于全量导 可重入阶段；如果在该值为no时重启任务导出流程不支持断点续传 示例值：yes

Table

数据同步库表信息描述

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
TableName	String	否	表名 示例值：t1
NewTableName	String	否	新表名 示例值：t1_new

名称	类型	必选	描述
FilterCondition	String	否	过滤条件 示例值：where id > 100
ColumnMode	String	否	是否同步表中所有列，All：当前表下的所有列,Partial(ModifySyncJobConfig接口里的对应字段ColumnMode暂不支持Partial)：当前表下的部分列，通过填充Columns字段详细表信息 示例值：All
Columns	Array of Column	否	同步的列信息，当ColumnMode为Partial时，必填
TmpTables	Array of String	否	同步临时表，注意此配置与NewTableName互斥，只能使用其中一种。当配置的同步对象为表级别且TableEditMode为pt时此项有意义，针对pt-osc等工具在同步过程中产生的临时表进行同步，需要提前将可能的临时表配置在这里，否则不会同步任何临时表。示例，如要对t1进行pt-osc操作，此项配置应该为["_t1_new","_t1_old"]；如要对t1进行gh-ost操作，此项配置应该为["_t1_ghc","_t1_gho","_t1_del"]，pt-osc与gh-ost产生的临时表可同时配置。 示例值：["_t1_old",_t1_new"]
TableEditMode	String	否	编辑表类型，rename(表映射)，pt(同步附加表) 示例值：pt

TableItem

表对象集合，当 TableMode 为 partial 时，此项需要填写

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
TableName	String	否	迁移的表名，大小写敏感 示例值：t1
NewTableName	String	否	迁移后的表名，当TableEditMode为rename时此项必填，注意此配置与TmpTables互斥，只能使用其中一种 示例值：t1_new
TmpTables	Array of String	否	迁移临时表，注意此配置与NewTableName互斥，只能使用其中一种。当配置的同步对象为表级别且TableEditMode为pt时此项有意义，针对pt-osc等工具在迁移过程中产生的临时表进行同步，需要提前将可能的临时表配置在这里，否则不会同步任何临时表。示例，如要对t1进行pt-osc操作，此项配置应该为["_t1_new","_t1_old"]；如要对t1进行gh-ost操作，此项配置应该为["_t1_ghc","_t1_gho","_t1_del"]，pt-osc与gh-ost产生的临时表可同时配置。 示例值：["_t1_old",_t1_new"]
TableEditMode	String	否	编辑表类型，rename(表映射)，pt(同步附加表) 示例值：rename

TagFilter

标签过滤

被如下接口引用：DescribeMigrationJobs, DescribeSubscribeJobs, DescribeSyncJobs。

名称	类型	必选	描述
TagKey	String	是	标签键值 示例值：key
TagValue	Array of String	否	标签值 示例值：[value1, value2]

TagItem

标签

被如下接口引用：CreateMigrationService, CreateSubscribe, CreateSyncJob, DescribeMigrationDetail, DescribeMigrationJobs, DescribeSubscribeDetail, DescribeSubscribeJobs, DescribeSyncJobs, ModifyMigrationJob。

名称	类型	必选	描述
TagKey	String	否	标签键 示例值：key
TagValue	String	否	标签值 示例值：value

TopicRule

单topic和自定义topic的描述。投递到单topic时，该数组的最后一项会被视为默认分区策略，所有未匹配到的数据都会按该策略投递，默认策略只支持 投递至partition0、按表名、表名+主键三种。

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
TopicName	String	否	topic名。单topic时，所有的TopicName必须相同 示例值：topicName
PartitionType	String	否	topic分区策略，自定义topic时支持：Random（随机投递），集中投递到单Topic时支持：AllInPartitionZero（全部投递至partition0）、PartitionByTable(按表名分区)、PartitionByTableAndKey(按表名加主键分区)、PartitionByCols(按列分区) 示例值：Random（自定义topic时使用），AllInPartitionZero（全部投递至partition0），PartitionByTable（按表名分区），PartitionByTableAndKey（按表名加主键分区）， PartitionByCols（按列名分区）
DbMatchMode	String	否	库名匹配规则，如Regular（正则匹配）, Default(不符合匹配规则的剩余库)，数组中最后一项必须为‘Default’ 示例值：Regular（正则匹配）, Default(不符合匹配规则的剩余库)
DbName	String	否	库名，DbMatchMode=Regular时生效 示例值：db1
TableMatchMode	String	否	表名匹配规则，如Regular（正则匹配）, Default(不符合匹配规则的剩余表)，数组中最后一项必须为‘Default’ 示例值：Regular（正则匹配）, Default(不符合匹配规则的剩余表)
TableName	String	否	表名，仅TableMatchMode=Regular时生效 示例值：tb1
Columns	Array of String	否	按列分区时需要选择配置列名，可以选择多列 注意：此字段可能返回 null，表示取不到有效值。 示例值：c1

TradeInfo

计费状态信息

被如下接口引用：DescribeMigrationDetail, DescribeMigrationJobs。

名称	类型	描述
DealName	String	交易订单号 示例值：2022071171500104792948 1
LastDealName	String	上一次交易订单号 示例值：2022071171500104792934 0
InstanceClass	String	实例规格，包括：micro、small、medium、large、xlarge、2xlarge等 示例值：xlarge
TradeStatus	String	计费任务状态， normal(计费或待计费)、resizing(变配中)、reversing(冲正中，比较短暂的状态)、isolating(隔离中，比较短暂的状态)、isolated(已隔离)、offlining(下线中)、offlined(已下线)、notBilled(未计费) 示例值：normal
ExpireTime	String	到期时间，格式为"yyyy-mm-dd hh:mm:ss" 示例值：2023-12-12 15:54:18
OfflineTime	String	下线时间，格式为"yyyy-mm-dd hh:mm:ss" 示例值：2023-12-19 15:54:18

名称	类型	描述
IsolateTime	String	隔离时间，格式为"yyyy-mm-dd hh:mm:ss" 示例值：2023-12-11 15:54:18
OfflineReason	String	下线原因 示例值：user_offline
IsolateReason	String	隔离原因 示例值：user_isolate
PayType	String	付费类型，包括：postpay(后付费)、prepay(预付费) 示例值：postpay
BillingType	String	任务计费类型，包括：billing(计费)、notBilling(不计费)、 promotions(促销活动中) 示例值：billing

View

数据同步view的描述

被如下接口引用：ConfigureSyncJob, DescribeSyncJobs, ModifySyncJobConfig。

名称	类型	必选	描述
ViewName	String	否	view名 示例值：v1
NewViewName	String	否	预留字段、目前暂时不支持view的重命名 示例值：v1_new

ViewItem

视图对象

被如下接口引用：DescribeMigrationDetail, ModifyMigrationJob。

名称	类型	必选	描述
ViewName	String	否	视图名称 示例值：t1_view
NewViewName	String	否	迁移后的视图名称 示例值：t1_view_new

错误码

最近更新时间：2024-12-20 01:26:06

功能说明

如果返回结果中存在 Error 字段，则表示调用 API 接口失败。例如：

```
{
"Response": {
"Error": {
"Code": "AuthFailure.SignatureFailure",
"Message": "The provided credentials could not be validated. Please check your signature is correct."
},
"RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
}
}
```

Error 中的 Code 表示错误码，Message 表示该错误的具体信息。

错误码列表

公共错误码

错误码	说明
ActionOffline	接口已下线。
AuthFailure.InvalidAuthorization	请求头部的 `Authorization` 不符合腾讯云标准。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。
AuthFailure.MFAFailure	MFA 错误。
AuthFailure.SecretIdNotFound	密钥不存在。请在 控制台 检查密钥是否已被删除或者禁用，如状态正常，请检查密钥是否填写正确，注意前后不得有空格。
AuthFailure.SignatureExpire	签名过期。Timestamp 和服务器时间相差不得超过五分钟，请检查本地时间是否和标准时间同步。
AuthFailure.SignatureFailure	签名错误。签名计算错误，请对照调用方式中的签名方法文档检查签名计算过程。
AuthFailure.TokenFailure	token 错误。
AuthFailure.UnauthorizedOperation	请求未授权。请参考 CAM 文档对鉴权的说明。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidAction	接口不存在。
InvalidParameter	参数错误（包括参数格式、类型等错误）。
InvalidParameterValue	参数取值错误。
InvalidRequest	请求 body 的 multipart 格式错误。
IpInBlacklist	IP 地址在黑名单中。
IpNotInWhitelist	IP 地址不在白名单中。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数。

错误码	说明
NoSuchProduct	产品不存在
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
RequestLimitExceeded.GlobalRegionUinLimitExceeded	主账号超过频率限制。
RequestLimitExceeded.IPLimitExceeded	IP 限频。
RequestLimitExceeded.UinLimitExceeded	主账号限频。
RequestSizeLimitExceeded	请求包超过限制大小。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResponseSizeLimitExceeded	返回包超过限制大小。
ServiceUnavailable	当前服务暂时不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误，用户多传未定义的参数会导致错误。
UnsupportedOperation	操作不支持。
UnsupportedProtocol	http(s) 请求协议错误，只支持 GET 和 POST 请求。
UnsupportedRegion	接口不支持所传地域。

业务错误码

错误码	说明
AuthFailure	CAM签名/鉴权错误。
AuthFailure.AuthFailureError	认证失败。
AuthFailure.AuthorizedOperationDenyError	认证拒绝错误。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
DryRunOperation.DryRunOperationError	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StartJobFailed	任务开始失败。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
FailedOperation.UserNotAuthed	用户没有实名认证。
InternalError.AddTaskError	添加异步任务失败。
InternalError.CeleryError	内部调度系统错误。
InternalError.CgwSystemError	cgw系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.ExecHttpRequestError	http请求出错。

版权所有：腾讯云计算（北京）有限责任公司

错误码	说明
InternalError.InternalErrorError	内部错误。
InternalError.InternalHttpServerError	http请求访问出错。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.InternalTradeError	调用计费服务失败。
InternalError.LockError	锁冲突。
InternalError.NotEnoughMoneyError	用户余额不足。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter.ControllerNotFoundError	异常错误导致当前接口未注册。
InvalidParameter.InstanceNotFound	实例不存在。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.DataConvertError	数据转换错误。
InvalidParameterValue.InvalidParameterValue	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.LimitExceededError	数量超出限制。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
MissingParameter.MissingParameterError	参数丢失。
OperationDenied	操作被拒绝。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.NotEnoughMoneyError	用户余额不足。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceInUse.ResourceInUseError	资源在使用中。
ResourceInsufficient.ResourceInsufficientError	资源短缺。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。
ResourceUnavailable.ResourceUnavailableError	资源不可用。
ResourcesSoldOut.ResourcesSoldOutError	资源售罄。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.PermissionDenied	操作未被授权。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnknownParameter.UnknownParameterError	未知参数。

版权所有：腾讯云计算（北京）有限责任公司

错误码	说明
UnsupportedOperation.IntraNetUserNotTaggedError	自研上云用户创建迁移任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。
UnsupportedOperation.IntraNetUserNotTaggedForSyncJobError	自研上云用户创建同步任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。
UnsupportedOperation.UnsupportedOperationError	不支持的操作。

数据传输服务 API 2018-03-30
产品版本

最近更新时间：2023-10-23 00:34:38

您正在浏览dts产品文档的版本: 2018-03-30

最新版本

- 2021-12-06

以往版本

- 2018-03-30 (当前版本)

更新历史

最近更新时间：2025-05-22 01:17:04

第 23 次发布

发布时间：2025-05-22 01:17:01

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DstInfo
  - **废弃成员：** Ip, Port

第 22 次发布

发布时间：2024-06-21 01:37:37

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DescribeRegionConf

删除数据结构：

- SubscribeRegionConf

第 21 次发布

发布时间：2024-06-13 01:40:12

本次发布包含了以下内容：

改善已有的文档。

预下线接口：

- DescribeRegionConf

第 20 次发布

发布时间：2024-06-07 01:43:53

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SubsErr
  - **修改成员：** Message

第 19 次发布

发布时间：2022-04-28 06:11:50

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeSubscribeConf
  - 新增出参：SubscribeVersion, Errors

新增数据结构：

- SubsErr

修改数据结构：

- MigrateJobInfo
  - 新增成员：SrcInfoMulti

第 18 次发布

发布时间：2022-04-26 06:09:43

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateSubscribe
  - 新增入参：Name

第 17 次发布

发布时间：2022-04-22 06:09:15

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateMigrateJob
  - 新增入参：SrcNodeType, SrcInfoMulti
- ModifyMigrateJob
  - 新增入参：SrcNodeType, SrcInfoMulti

第 16 次发布

发布时间：2021-12-21 08:05:11

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateMigrateJob
  - 新增入参：Tags

第 15 次发布

发布时间：2021-11-09 08:10:46

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DescribeSyncCheckJob
- ModifySyncJob

删除数据结构：

- SyncCheckStepInfo
- SyncOption

第 14 次发布

发布时间：2021-11-08 08:05:43

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- CreateSyncCheckJob
- CreateSyncJob
- DeleteSyncJob
- DescribeSyncJobs
- StartSyncJob
- SwitchDrToMaster

删除数据结构：

- SyncDetailInfo
- SyncInstanceInfo
- SyncJobInfo
- SyncStepDetailInfo

第 13 次发布

发布时间：2021-09-02 08:04:47

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeMigrateJobs
  - 新增入参：TagFilters

修改数据结构：

- DstInfo
  - **修改成员：** InstanceId
- MigrateJobInfo
  - 新增成员：Tags

第 12 次发布

发布时间：2021-06-16 08:05:20

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CompleteMigrateJob
  - 新增入参：CompleteMode

第 11 次发布

发布时间：2021-02-03 08:04:31

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeSubscribes
  - 新增入参：SubscribeVersion

修改数据结构：

- SubscribeInfo
  - 新增成员：SubscribeVersion

第 10 次发布

发布时间：2020-08-28 08:03:00

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ModifySubscribeAutoRenewFlag

修改接口：

- CreateSubscribe
  - 新增入参：Tags
- DescribeSubscribeConf
  - 新增出参：Tags, AutoRenewFlag
- DescribeSubscribes
  - 新增入参：TagFilters

新增数据结构：

- TagFilter
- TagItem

修改数据结构：

- SubscribeInfo
  - 新增成员：Tags, AutoRenewFlag

第 9 次发布

发布时间：2020-04-29 08:09:34

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DstInfo
  - 新增成员：User, Password

第 8 次发布

发布时间：2020-03-20 08:07:26

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ActivateSubscribe
- CreateSubscribe
- DescribeAsyncRequestInfo
- DescribeRegionConf
- DescribeSubscribeConf
- DescribeSubscribes
- IsolateSubscribe
- ModifySubscribeConsumeTime

- ModifySubscribeName
- ModifySubscribeObjects
- ModifySubscribeVipVport
- OfflineIsolatedSubscribe
- ResetSubscribe

新增数据结构：

- SubscribeInfo
- SubscribeObject
- SubscribeRegionConf

第 7 次发布

发布时间：2020-03-19 08:07:50

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- ErrorInfo

修改数据结构：

- MigrateJobInfo
  - 新增成员：ErrorInfo

第 6 次发布

发布时间：2020-03-16 08:09:01

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- MigrateStepDetailInfo
  - 新增成员：StartTime

第 5 次发布

发布时间：2019-10-17 19:45:33

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SrcInfo
  - 新增成员：EngineVersion

第 4 次发布

发布时间：2019-08-16 16:50:11

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DstInfo
  - **修改成员：** Region

第 3 次发布

发布时间：2019-08-08 23:41:19

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- SrcInfo
  - 新增成员：CcnId

第 2 次发布

发布时间：2018-07-26 12:26:57

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateSyncCheckJob
- CreateSyncJob
- DeleteSyncJob
- DescribeSyncCheckJob
- DescribeSyncJobs
- ModifySyncJob
- StartSyncJob
- SwitchDrToMaster

新增数据结构：

- SyncCheckStepInfo
- SyncDetailInfo
- SyncInstanceInfo
- SyncJobInfo
- SyncOption
- SyncStepDetailInfo

第 1 次发布

发布时间：2018-06-29 17:14:44

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CompleteMigrateJob
- CreateMigrateCheckJob
- CreateMigrateJob
- DeleteMigrateJob
- DescribeMigrateCheckJob
- DescribeMigrateJobs
- ModifyMigrateJob
- StartMigrateJob
- StopMigrateJob

新增数据结构：

- ConsistencyParams
- DstInfo
- MigrateDetailInfo

- MigrateJobInfo
- MigrateOption
- MigrateStepDetailInfo
- SrcInfo

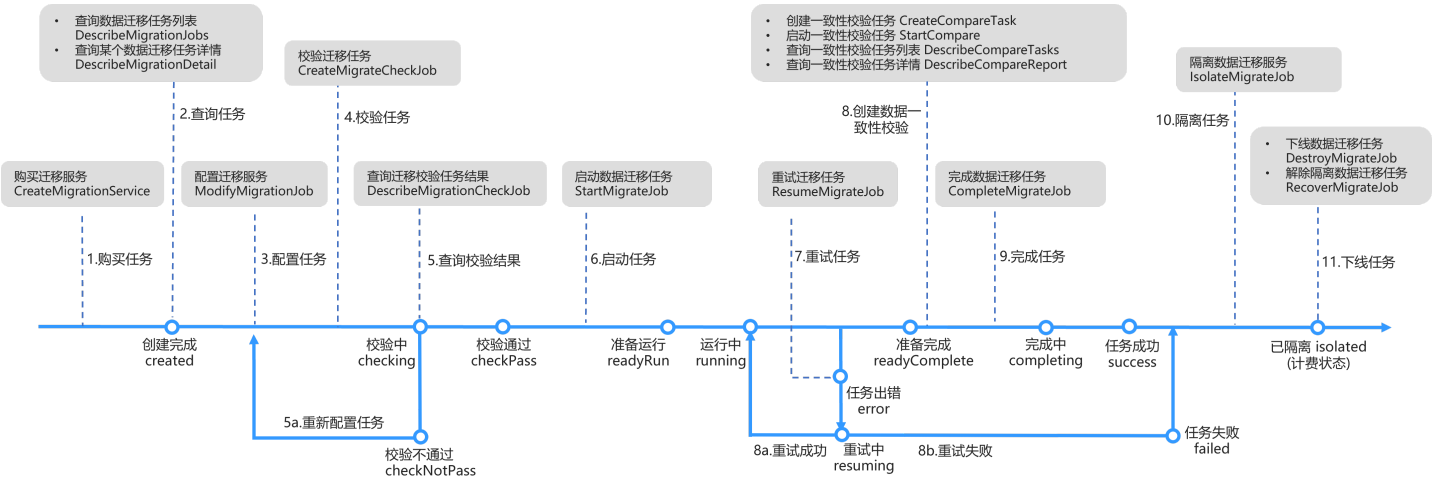
简介

最近更新时间：2025-04-25 01:29:06

数据传输服务（Data Transfer Service，DTS）提供数据迁移、数据同步、数据订阅于一体的数据库数据传输服务，帮助您在业务不停服的前提下轻松完成数据库迁移，利用实时同步通道轻松构建异地容灾的高可用数据库架构。

数据迁移 API 使用关键流程

数据迁移任务中使用的关键接口如下。圆圈表示迁移任务的不同状态，请按照图中序号，依次进行接口调用操作。



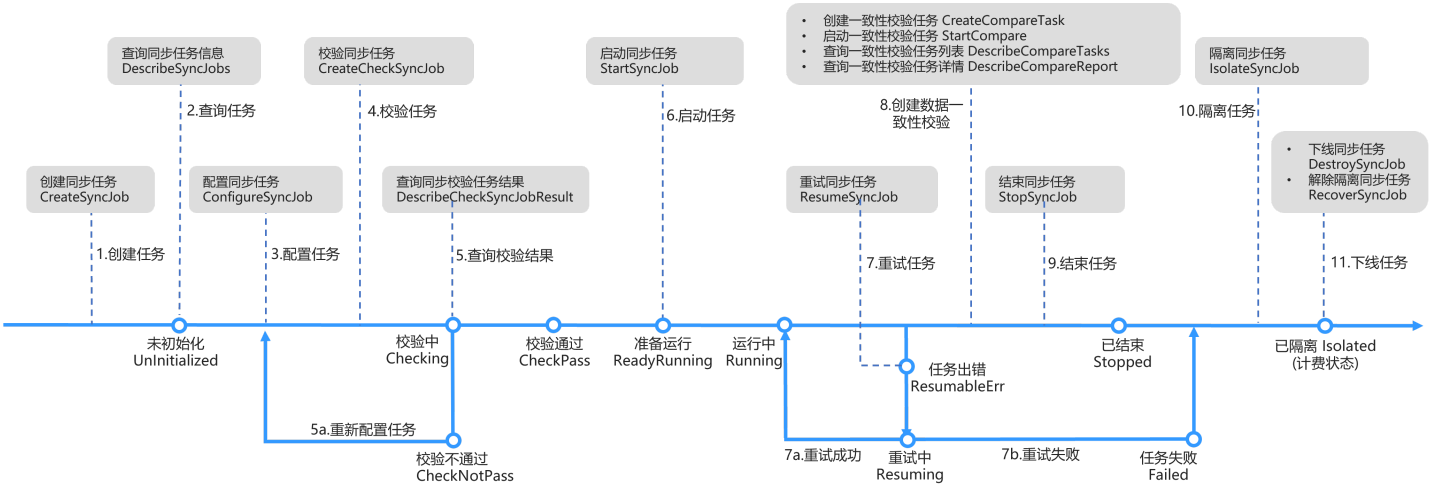
迁移任务中，各任务状态下支持的操作和相关接口如下表所示：

任务状态	支持的操作	关键接口	说明
–	–	购买迁移服务CreateMigrationService	购买迁移任务资源。购买完成后，会返回随机生成的迁移任务ID列表。
创建完成 created	配置	配置迁移服务 ModifyMigrationJob 校验迁移任务 CreateMigrateCheckJob 查询数据迁移任务列表 DescribeMigrationJobs	购买操作完成，下一步进行迁移任务配置。 任务配置完成后，发起校验。任务校验即在迁移任务启动前，对源库和目标库的各项参数进行校验，校验通过后才能启动迁移任务。 可通过“查询数据迁移任务列表 DescribeMigrationJobs”查询所有迁移任务列表，获取任务ID。
校验中 checking	–	查询迁移校验任务结果 DescribeMigrationCheckJob	迁移任务校验进行中。
校验通过 checkPass	立即启动、配置、校验	启动数据迁移任务 StartMigrateJob	迁移任务校验通过，启动数据迁移任务。
校验不通过 checkNotPass	配置、校验	配置迁移服务 ModifyMigrationJob	迁移任务校验不通过，需要重新配置迁移任务，然后再次发起校验。
准备运行 readyRun	终止	–	进行**启动**操作后，准备启动数据迁移任务的过渡状态。
运行中 running	终止、创建一致性校验任务（仅支持该功能的链路可操作）	终止数据迁移任务 StopMigrateJob	任务正在运行中。 用户如果不需要继续迁移，可以通过**终止**操作来取消任务。任务终止后不可恢复。
终止中 stopping	–	–	进行**终止**操作后的过渡状态。
已终止 canceled	–	隔离数据迁移服务 IsolateMigrateJob	已终止的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
准备完成 readyComplete	完成、终止、创建一致性校验任务（仅支持该功能的链路可操作）	完成数据迁移任务 CompleteMigrateJob 创建一致性校验任务 CreateCompareTask 启动一致性校验任务 StartCompare 查询一致性校验任务列表 DescribeCompareTasks 查询一致性校验任务详情 DescribeCompareReport	选择“全量+增量迁移“场景，当任务状态为”准备完成“时，表明全量和增量数据都已迁移完成，需要用户选择一个合适的时间，通过**完成**操作来结束任务。 建议在**完成**操作前先进行数据一致性校验。 数据一致性校验，即对比源库中选择迁移的数据，和已迁移到目标库中的数据结果是否一致，并查看不一致的数据详情。创建一致性校验任务成功后，会返回一致性校验的任务 ID。

任务状态	支持的操作	关键接口	说明
完成中 completing	–	–	进行**完成**操作后的过渡状态。
任务成功 success	–	隔离数据迁移服务 IsolateMigrateJob	迁移任务顺利完成。已完成的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
任务出错 error	终止、重试	重试迁移任务 ResumeMigrateJob	迁移过程中，出现异常导致任务中断，任务状态为“error”时，可以进行任务重试。
重试中 resuming	–	–	进行**重试**操作后的过渡状态，重试成功则任务正常进行，重试失败则任务状态为“任务失败 failed”。
任务失败 failed	重试（仅 Redis 迁移可进行重试操作）	隔离数据迁移服务 IsolateMigrateJob	迁移任务失败。 失败后的任务需要进行隔离操作，以便及时解冻费用（按量计费会冻结一小时费用）。
以上所有状态	查看、变配、隔离	查询某个数据迁移任务详情 DescribeMigrationDetail 调整实例规格 ModifyMigrateJobSpec 隔离数据迁移服务 IsolateMigrateJob	通过**查看**操作查看某个指定迁移任务的详情。 通过**变配**操作调整迁移链路的规格，不同规格的传输速率不同。 通过**隔离**操作将已结束的任务、失败的任务、不需要的任务，进行隔离。
已隔离 isolated (计费状态)	解隔离（恢复）、下线	解除隔离数据迁移任务 RecoverMigrateJob 下线数据迁移任务 DestroyMigrateJob	用户主动发起隔离操作，或者任务在欠费1天后，都会进入隔离状态，隔离中的资源会保留7天，业务停服不可用，隔离7天后资源会被清除。 已隔离的任务，用户可以主动发起下线操作，也可以等隔离7天后系统自动下线。 如果用户误操作了隔离，可以通过**解隔离**操作，将任务恢复到隔离前的状态。

数据同步 API 使用关键流程

数据同步任务中使用的关键接口如下。圆圈表示同步任务的不同状态，请按照图中序号，依次进行接口调用操作。



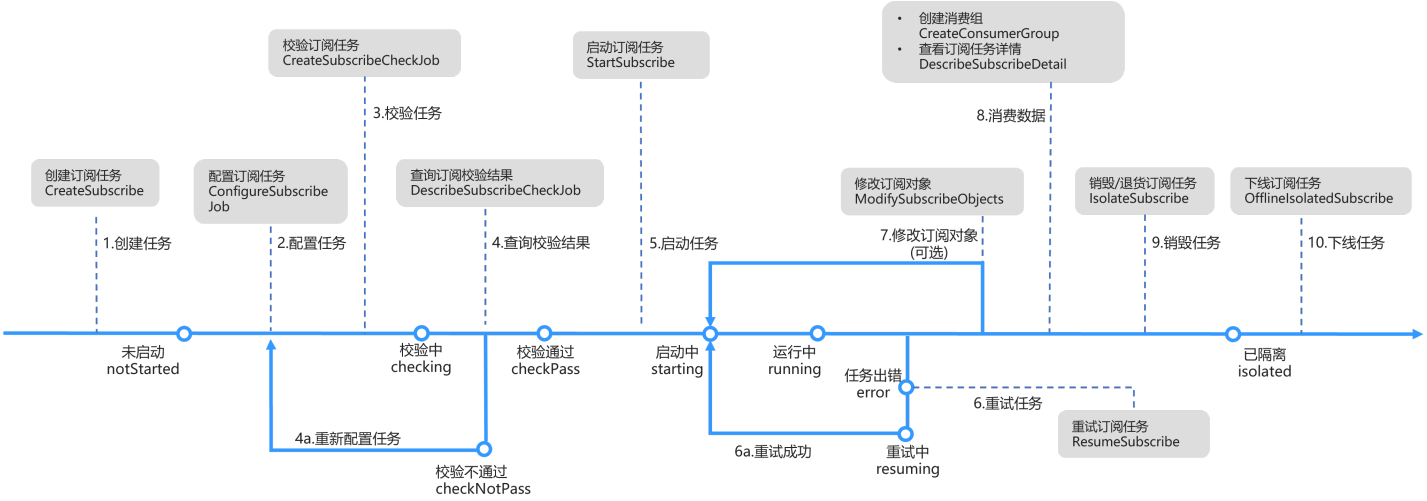
同步任务中，各任务状态下支持的操作和相关接口如下表所示：

任务状态	支持的操作	关键接口	说明
–	–	创建同步服务 CreateSyncJob	购买同步任务资源。购买完成后，会返回随机生成的同步任务ID列表。
未初始化 UnInitialized	配置	配置同步任务 ConfigureSyncJob 校验同步任务 CreateCheckSyncJob 查询同步任务信息 DescribeSyncJobs	购买操作完成，下一步进行同步任务配置。 任务配置完成后，发起校验。任务校验即在同步任务启动前，对源库和目标库的各项参数进行校验，校验通过后才能启动同步任务。 可通过“查询同步任务信息 DescribeSyncJobs”查询所有同步任务列表，获取任务ID。
校验中 Checking	–	查询同步校验任务结果 DescribeCheckSyncJobResult	同步任务校验进行中。
校验通过 CheckPass	启动、配置、校验	启动同步任务 StartSyncJob	同步任务校验通过，启动数据同步任务。

任务状态	支持的操作	关键接口	说明
校验不通过 CheckNotPass	配置、校验	配置同步任务 ConfigureSyncJob	同步任务校验不通过，需要重新配置同步任务，然后再次发起校验。
准备运行 ReadyRunning	结束	–	启动操作后，准备启动数据同步任务的过渡状态。
运行中 Running	结束	结束同步任务 StopSyncJob	任务正在运行中。 用户如果不需要继续同步，可以通过**结束**操作来停止任务。
结束中 Stopping	–	–	进行**结束**操作后的过渡状态。
已结束 Stopped	–	隔离同步任务 IsolateSyncJob	同步任务顺利完成。已完成的任务需要进行隔离，以便及时解冻费用（按量计费会冻结一小时费用）。
任务出错 ResumableErr	结束、重试	重试同步任务 ResumeSyncJob	同步过程中，出现异常导致任务中断，任务状态为“ResumableErr”时，可以进行任务重试。
重试中 Resuming	–	–	进行**重试**操作后的过渡状态，重试成功则任务正常进行，重试失败则任务状态为“任务失败 Failed”。
任务失败 Failed	–	隔离同步任务 IsolateSyncJob	同步任务失败。 失败后的任务需要进行隔离操作，以便及时解冻费用（按量计费会冻结一小时费用）。
以上所有状态	查看、变配、隔离	查询同步任务信息 DescribeSyncJobs 调整同步任务规格 ResizeSyncJob 隔离同步任务 IsolateSyncJob	通过**查看**操作查看某个指定同步任务的详情。 通过**变配**操作调整同步链路的规格，不同规格的传输速率不同。 通过**隔离**操作将已结束的任务、失败的任务、不需要的任务，进行隔离。
已隔离 Isolated (计费状态)	解隔离（恢复）、下线	解除隔离同步任务 RecoverSyncJob 下线同步任务 DestroySyncJob	用户主动发起隔离操作，或者任务在欠费1天后，都会进入隔离状态，隔离中的资源会保留7天，业务停服不可用，隔离7天后资源会被清除。 已隔离的任务，用户可以主动发起下线操作，也可以等隔离7天后系统自动下线。 如果用户误操作了隔离，可以通过**解隔离**操作，将任务恢复到隔离前的状态。

数据订阅 API 使用关键流程

数据订阅任务中使用的关键接口如下。圆圈表示订阅任务的不同状态，请按照图中序号，依次进行接口调用操作。



任务状态	支持的操作	关键接口	说明
–	–	创建订阅任务 CreateSubscribe	购买订阅任务资源。购买完成后，会返回随机生成的订阅任务ID列表。
未启动 notStarted	配置订阅、销毁/退货	配置订阅任务 ConfigureSubscribe 校验订阅任务 CreateSubscribeCheckJob	购买完成，下一步进行订阅任务配置。 任务配置完成后，发起校验。校验任务即在订阅任务启动前，对源库的各项参数进行校验，校验通过后才能启动订阅任务。
校验中 checking		查询订阅校验结果	订阅任务校验进行中。

任务状态	支持的操作	关键接口	说明
		DescribeSubscribeCheckJob	
校验通过 checkPass	启动、配置订阅、校验、销毁/退货	启动校验任务 StartSubscribe	校验通过，启动数据订阅任务。
校验不通过 checkNotPass	配置订阅、校验、销毁/退货	配置订阅任务 ConfigureSubscribeJob	订阅任务校验不通过，需要重新配置订阅任务，然后再次发起校验。
启动中 starting	销毁/退货	–	订阅任务启动中，过渡状态。
任务运行 running	修改订阅对象、重置订阅、销毁/退货、查看任务详情	创建消费组 CreateConsumerGroup 修改订阅对象 ModifySubscribeObjects 重置订阅任务 ResetSubscribe 销毁/退货订阅任务 IsolateSubscribe 查看订阅任务详情 DescribeSubscribeDetail	订阅任务运行中，此时可以创建消费组消费Kafka。 修改订阅对象（ModifySubscribeObjects），是对正在进行的任务，追加或者删除对象，Kafka 分区规则等。 销毁任务，是停止任务并清空配置和kafka数据，同时进行计费清算。 重置订阅任务，是对不再需要的任务进行重置，旧的任务会停止并清空历史配置和kafka数据，然后重新配置新任务。重置任务可以避免用户先退费旧任务，再购买新任务，频繁地进行购买操作。 任务运行中可查看任务详情并操作kafka相关接口。
停止中 stopping	–		重置订阅对象时，订阅任务停止中。重置后待任务状态变为“未启动 notStarted”状态才可重新配置任务。
任务出错 error	恢复、销毁/退货	恢复订阅任务 ResumeSubscribe	订阅过程中，如果任务出现异常导致任务中断，状态为“error”时，可以尝试恢复任务。
已隔离 isolated	下线	下线已隔离的订阅任务 OfflineIsolatedSubscribe	销毁/退货操作后，任务会进入隔离的状态。隔离的任务，用户可以主动发起下线操作，也可以等隔离1天后系统自动下线。

API 概览

最近更新时间：2024-08-02 01:32:29

数据传输服务相关接口

接口名称	接口功能	频率限制（次/秒）
CompleteMigrateJob	完成数据迁移任务	20
CreateMigrateCheckJob	创建校验迁移任务	20
CreateMigrateJob	创建数据迁移任务	20
DeleteMigrateJob	删除数据迁移任务	20
DescribeMigrateCheckJob	获取迁移校验结果	20
DescribeMigrateJobs	查询数据迁移任务	100
ModifyMigrateJob	修改数据迁移任务	20
StartMigrateJob	启动数据迁移任务	20
StopMigrateJob	撤销数据迁移任务	20

数据订阅相关接口

接口名称	接口功能	频率限制（次/秒）
ActivateSubscribe	配置数据订阅通道	20
CreateSubscribe	创建订阅对象	20
DescribeAsyncRequestInfo	查询任务执行结果	20
DescribeSubscribeConf	查询订阅实例配置	20
DescribeSubscribes	获取数据订阅实例列表	100
IsolateSubscribe	隔离订阅实例	20
ModifySubscribeAutoRenewFlag	修改订阅实例自动续费标识	20
ModifySubscribeConsumeTime	修改数据订阅实例通道的消费时间点	20
ModifySubscribeName	修改数据订阅实例的名称	20
ModifySubscribeObjects	修改数据订阅通道的订阅规则	20
ModifySubscribeVipVport	修改数据订阅实例的IP和端口号	20
OfflineIsolatedSubscribe	下线已隔离的数据订阅实例	20
ResetSubscribe	重置数据订阅实例	20

注意：

以上给出的接口频率限制维度为 `API` + 接入地域 + 子账号，有关限频更多说明参考：API 频率限制说明

调用方式
请求结构

最近更新时间：2025-05-21 01:17:45

1. 服务地址

API 支持就近地域接入，本产品就近地域接入域名为 dts.tencentcloudapi.com ，也支持指定地域域名访问，例如广州地域的域名为 dts.ap-guangzhou.tencentcloudapi.com 。

推荐使用就近地域接入域名。根据调用接口时客户端所在位置，会自动解析到**最近的**某个具体地域的服务器。例如在广州发起请求，会自动解析到广州的服务器，效果和指定 dts.ap-guangzhou.tencentcloudapi.com 是一致的。

注意：对时延敏感的业务，建议指定带地域的域名。

注意：域名是 API 的接入点，并不代表产品或者接口实际提供服务的地域。产品支持的地域列表请在调用方式/公共参数文档中查阅，接口支持的地域请在接口文档输入参数中查阅。

目前支持的域名列表为：

接入地域	域名
就近地域接入（推荐，只支持非金融区）	dts.tencentcloudapi.com
华南地区（广州）	dts.ap-guangzhou.tencentcloudapi.com
华东地区（上海）	dts.ap-shanghai.tencentcloudapi.com
华东地区（南京）	dts.ap-nanjing.tencentcloudapi.com
华北地区（北京）	dts.ap-beijing.tencentcloudapi.com
西南地区（成都）	dts.ap-chengdu.tencentcloudapi.com
西南地区（重庆）	dts.ap-chongqing.tencentcloudapi.com
港澳台地区（中国香港）	dts.ap-hongkong.tencentcloudapi.com
亚太东南（新加坡）	dts.ap-singapore.tencentcloudapi.com
亚太东南（雅加达）	dts.ap-jakarta.tencentcloudapi.com
亚太东南（曼谷）	dts.ap-bangkok.tencentcloudapi.com
亚太东北（首尔）	dts.ap-seoul.tencentcloudapi.com
亚太东北（东京）	dts.ap-tokyo.tencentcloudapi.com
美国东部（弗吉尼亚）	dts.na-ashburn.tencentcloudapi.com
美国西部（硅谷）	dts.na-siliconvalley.tencentcloudapi.com
南美地区（圣保罗）	dts.sa-saopaulo.tencentcloudapi.com
欧洲地区（法兰克福）	dts.eu-frankfurt.tencentcloudapi.com

注意：由于金融区和非金融区是隔离不互通的，因此当访问金融区服务时（公共参数 Region 为金融区地域），需要同时指定带金融区地域的域名，最好和 Region 的地域保持一致。

金融区接入地域	金融区域名
华东地区（上海金融）	dts.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	dts.ap-shenzhen-fsi.tencentcloudapi.com

2. 通信协议

腾讯云 API 的所有接口均通过 HTTPS 进行通信，提供高安全性的通信通道。

3. 请求方法

支持的 HTTP 请求方法:

- POST（推荐）
- GET

POST 请求支持的 Content-Type 类型:

- application/json（推荐），必须使用签名方法 v3（TC3-HMAC-SHA256）。
- application/x-www-form-urlencoded，必须使用签名方法 v1（HmacSHA1 或 HmacSHA256）。
- multipart/form-data（仅部分接口支持），必须使用签名方法 v3（TC3-HMAC-SHA256）。

GET 请求的请求包大小不得超过32KB。POST 请求使用签名方法 v1（HmacSHA1、HmacSHA256）时不得超过1MB。POST 请求使用签名方法 v3（TC3-HMAC-SHA256）时支持10MB。

4. 字符编码

均使用 `UTF-8` 编码。

公共参数

最近更新时间：2025-03-28 01:31:25

公共参数是用于标识用户和接口签名的参数，如非必要，在每个接口单独的文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

公共参数的具体内容会因您使用的签名方法版本不同而有所差异。

使用签名方法 v3 的公共参数

签名方法 v3（有时也称作 TC3-HMAC-SHA256）相比签名方法 v1（有些文档可能会简称签名方法），更安全，支持更大的请求包，支持 POST JSON 格式，性能有一定提升，推荐使用该签名方法计算签名。完整介绍详见 签名方法 v3。

注意：出于简化的目的，部分接口文档中的示例使用的是签名方法 v1 GET 请求，而不是更安全的签名方法 v3。

使用签名方法 v3 时，公共参数需要统一放到 HTTP Header 请求头部中，如下表所示：

参数名称	类型	必选	描述
Action	String	是	HTTP 请求头：X-TC-Action。操作的接口名称。取值参考接口文档输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	-	HTTP 请求头：X-TC-Region。地域参数，用来标识希望操作哪个地域的数据。取值参考接口文档中输入参数章节关于公共参数 Region 的说明。**注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。**
Timestamp	Integer	是	HTTP 请求头：X-TC-Timestamp。当前 UNIX 时间戳，可记录发起 API 请求的时间。例如 1529223702。**注意：如果与服务器时间相差超过5分钟，会引起签名过期错误。**
Version	String	是	HTTP 请求头：X-TC-Version。操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
Authorization	String	是	HTTP 标准身份认证头部字段，例如： TC3-HMAC-SHA256 Credential=AKID***/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 其中， - TC3-HMAC-SHA256：签名方法，目前固定取该值； - Credential：签名凭证，AKID*** 是 SecretId；Date 是 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为具体产品名，通常为域名前缀。例如，域名 cvm.tencentcloudapi.com 意味着产品名是 cvm。本产品取值为 dts；tc3_request 为固定字符串； - SignedHeaders：参与签名计算的头部信息，content-type 和 host 为必选头部； - Signature：签名摘要，计算过程详见 文档。
Token	String	否	HTTP 请求头：X-TC-Token。即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	HTTP 请求头：X-TC-Language。指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表中的前十个，接口参数设置为偏移量 Offset=0，返回数量 Limit=10，则其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: ap-guangzhou
```

HTTP POST（application/json）请求结构示例：

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: application/json
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

{"Offset":0,"Limit":10}
```

HTTP POST（multipart/form-data）请求结构示例（仅特定的接口支持）：

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: multipart/form-data; boundary=58731222010402
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

--58731222010402
Content-Disposition: form-data; name="Offset"

0
--58731222010402
Content-Disposition: form-data; name="Limit"

10
--58731222010402--
```

使用签名方法 v1 的公共参数

使用签名方法 v1（有时会称作 HmacSHA256 和 HmacSHA1），公共参数需要统一放到请求串中，完整介绍详见文档

参数名称	类型	必选	描述
Action	String	是	操作的接口名称。取值参考接口文档中输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	-	地域参数，用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。**注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。**
Timestamp	Integer	是	当前 UNIX 时间戳，可记录发起 API 请求的时间。例如1529223702，如果与当前时间相差过大，会引起签名过期错误。
Nonce	Integer	是	随机正整数，与 Timestamp 联合起来，用于防止重放攻击。
SecretId	String	是	在 云API密钥 上申请的标识身份的 SecretId，一个 SecretId 对应唯一的 SecretKey ，而 SecretKey 会用来生成请求签名 Signature。
Signature	String	是	请求签名，用来验证此次请求的合法性，需要用户根据实际的输入参数计算得出。具体计算方法参见 文档。

参数名称	类型	必选	描述
Version	String	是	操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
SignatureMethod	String	否	签名方式，目前支持 HmacSHA256 和 HmacSHA1。只有指定此参数为 HmacSHA256 时，才使用 HmacSHA256 算法验证签名，其他情况均使用 HmacSHA1 验证签名。
Token	String	否	即 安全凭证服务 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。
Language	String	否	指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。

假设用户想要查询广州地域的云服务器实例列表，其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID********************************

Host: cvm.tencentcloudapi.com
```

HTTP POST 请求结构示例：

```
https://cvm.tencentcloudapi.com/

Host: cvm.tencentcloudapi.com
Content-Type: application/x-www-form-urlencoded

Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****************************
```

地域列表

本产品所有接口 Region 字段的可选值如下表所示。如果接口不支持该表中的所有地域，则会在接口文档中单独说明。

地域	取值
亚太东南（曼谷）	ap-bangkok
华北地区（北京）	ap-beijing
西南地区（成都）	ap-chengdu
西南地区（重庆）	ap-chongqing
华南地区（广州）	ap-guangzhou
港澳台地区（中国香港）	ap-hongkong
亚太东南（雅加达）	ap-jakarta
华东地区（南京）	ap-nanjing
亚太东北（首尔）	ap-seoul
华东地区（上海）	ap-shanghai
华东地区（上海金融）	ap-shanghai-fsi
华南地区（深圳金融）	ap-shenzhen-fsi
亚太东南（新加坡）	ap-singapore

地域	取值
亚太东北（东京）	ap-tokyo
欧洲地区（法兰克福）	eu-frankfurt
美国东部（弗吉尼亚）	na-ashburn
美国西部（硅谷）	na-siliconvalley
南美地区（圣保罗）	sa-saopaulo

版权所有：腾讯云计算（北京）有限责任公司

签名方法 v3

最近更新时间：2024-12-25 01:31:35

以下文档说明了签名方法 v3 的签名过程，但仅在您编写自己的代码来调用腾讯云 API 时才有用。我们推荐您使用 腾讯云 API Explorer ，腾讯云 SDK 和 腾讯云命令行工具（TCCLI）等开发者工具，从而无需学习如何对 API 请求进行签名。

推荐使用 API Explorer　　点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个请求进行身份验证，用户需要使用安全凭证，经过特定的步骤对请求进行签名（Signature），每个请求都需要在公共参数中指定该签名结果并以指定的方式和格式发送请求。

为什么要进行签名

签名通过以下方式帮助保护请求：

1. 验证请求者的身份
   签名确保请求是由持有有效访问密钥的人发送的。请参阅控制台 云 API 密钥 页面获取密钥相关信息。

2. 保护传输中的数据
   为了防止请求在传输过程中被篡改，腾讯云 API 会使用请求参数来计算请求的哈希值，并将生成的哈希值加密后作为请求的一部分，发送到腾讯云 API 服务器。服务器会使用收到的请求参数以同样的过程计算哈希值，并验证请求中的哈希值。如果请求被篡改，将导致哈希值不一致，腾讯云 API 将拒绝本次请求。

签名方法 v3（TC3-HMAC-SHA256）功能上覆盖了以前的签名方法 v1，而且更安全，支持更大的请求，支持 JSON 格式，POST 请求支持传空数组和空字符串，性能有一定提升，推荐使用该签名方法计算签名。

首次接触，建议使用 API Explorer 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v3”，可以对生成签名过程进行验证，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 Python、Java、PHP、Go、NodeJS、.NET、C++、Ruby。

申请安全凭证

本文使用的安全凭证为密钥，密钥包括 SecretId 和 SecretKey。每个用户最多可以拥有两对密钥。

- SecretId：用于标识 API 调用者身份，可以简单类比为用户名。
- SecretKey：用于验证 API 调用者的身份，可以简单类比为密码。
- **用户必须严格保管安全凭证，避免泄露，否则将危及财产安全。如已泄露，请立刻禁用该安全凭证。**

申请安全凭证的具体步骤如下：

1. 登录 腾讯云管理中心控制台 。
2. 前往 云API密钥 的控制台页面。
3. 在 云API密钥 页面，单击【新建密钥】创建一对密钥。

签名版本 v3 签名过程

云 API 支持 GET 和 POST 请求。对于GET方法，只支持 `Content-Type: application/x-www-form-urlencoded` 协议格式。对于POST方法，目前支持 `Content-Type: application/json` 以及 `Content-Type: multipart/form-data` 两种协议格式，json 格式绝大多数接口均支持，multipart 格式只有特定接口支持，此时该接口不能使用 json 格式调用，参考具体业务接口文档说明。推荐使用 POST 请求，因为两者的结果并无差异，但 GET 请求只支持 32 KB 以内的请求包。

下面以云服务器查询广州区实例列表作为例子，分步骤介绍签名的计算过程。我们选择该接口是因为：

1. 云服务器默认已开通，该接口很常用；
2. 该接口是只读的，不会改变现有资源的状态；
3. 接口覆盖的参数种类较全，可以演示包含数据结构的数组如何使用。

在示例中，不论公共参数或者接口的参数，我们尽量选择容易犯错的情况。在实际调用接口时，请根据实际情况来，每个接口的参数并不相同，不要照抄这个例子的参数和值。此外，这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数（在 HTTP 头部设置，添加 X-TC- 前缀）。

假设用户的 SecretId 和 SecretKey 分别是：`AKID********************************` 和 `********************************`。用户想查看广州区云服务器名为“未命名”的主机状态，只返回一条数据。则请求可能为：

```
curl -X POST https://cvm.tencentcloudapi.com \
-H "Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f" \
-H "Content-Type: application/json; charset=utf-8" \
-H "Host: cvm.tencentcloudapi.com" \
-H "X-TC-Action: DescribeInstances" \
-H "X-TC-Timestamp: 1551113065" \
-H "X-TC-Version: 2017-03-12" \
-H "X-TC-Region: ap-guangzhou" \
-d '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
```

下面详细解释签名计算过程。

1. 拼接规范请求串

按如下伪代码格式拼接规范请求串（CanonicalRequest）：

```
CanonicalRequest =
HTTPRequestMethod + '\n' +
CanonicalURI + '\n' +
CanonicalQueryString + '\n' +
CanonicalHeaders + '\n' +
SignedHeaders + '\n' +
HashedRequestPayload
```

字段名称	解释
HTTPRequestMethod	HTTP 请求方法（GET、POST ）。此示例取值为 `POST`。
CanonicalURI	URI 参数，API 3.0 固定为正斜杠（/）。
CanonicalQueryString	发起 HTTP 请求 URL 中的查询字符串，对于 POST 请求，固定为空字符串""，对于 GET 请求，则为 URL 中问号（?）后面的字符串内容，例如：Limit=10&Offset=0。 注意：CanonicalQueryString 需要参考 RFC3986 进行 URLEncode 编码（特殊字符编码后需大写字母），字符集 UTF-8。推荐使用编程语言标准库进行编码。
CanonicalHeaders	参与签名的头部信息，至少包含 host 和 content-type 两个头部，也可加入其他头部参与签名以提高自身请求的唯一性和安全性，此示例额外增加了接口名头部。 拼接规则： 1. 头部 key 和 value 统一转成小写，并去掉首尾空格，按照 key:value\n 格式拼接； 2. 多个头部，按照头部 key（小写）的 ASCII 升序进行拼接。 此示例计算结果是 `content-type:application/json; charset=utf-8\nhost:cvm.tencentcloudapi.com\nx-tc-action:describeinstances\n`。 注意：content-type 必须和实际发送的相符合，有些编程语言网络库即使未指定也会自动添加 charset 值，如果签名时和发送时不一致，服务器会返回签名校验失败。
SignedHeaders	参与签名的头部信息，说明此次请求有哪些头部参与了签名，和 CanonicalHeaders 包含的头部内容是一一对应的。content-type 和 host 为必选头部。 拼接规则： 1. 头部 key 统一转成小写； 2. 多个头部 key（小写）按照 ASCII 升序进行拼接，并且以分号（;）分隔。 此示例为 content-type;host;x-tc-action
HashedRequestPayload	请求正文（payload，即 body，此示例为 `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`）的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(RequestPayload)))，即对 HTTP 请求正文做 SHA256 哈希，然后十六进制编

字段名称	解释
	码，最后编码串转换成小写字母。对于 GET 请求，RequestPayload 固定为空字符串。此示例计算结果是 35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064。

根据以上规则，示例中得到的规范请求串如下：

```
POST
/

content-type:application/json; charset=utf-8
host:cvm.tencentcloudapi.com
x-tc-action:describeinstances

content-type;host;x-tc-action
35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064
```

2. 拼接待签名字符串

按如下格式拼接待签名字符串：

```
StringToSign =
Algorithm + "\n" +
RequestTimestamp + "\n" +
CredentialScope + "\n" +
HashedCanonicalRequest
```

字段名称	解释
Algorithm	签名算法，目前固定为 TC3-HMAC-SHA256。
RequestTimestamp	请求时间戳，即请求头部的公共参数 X-TC-Timestamp 取值，取当前时间 UNIX 时间戳，精确到秒。此示例取值为 1551113065。
CredentialScope	凭证范围，格式为 Date/service/tc3_request，包含日期、所请求的服务和终止字符串（tc3_request）。**Date 为 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致**；service 为产品名，必须与调用的产品域名一致。此示例计算结果是 2019-02-25/cvm/tc3_request。
HashedCanonicalRequest	前述步骤拼接所得规范请求串的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(CanonicalRequest)))。此示例计算结果是 7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84。

注意：

1. Date 必须从时间戳 X-TC-Timestamp 计算得到，且时区为 UTC+0。如果加入系统本地时区信息，例如东八区，将导致白天和晚上调用成功，但是凌晨时调用必定失败。假设时间戳为 1551113065，在东八区的时间是 2019-02-26 00:44:25，但是计算得到的 Date 取 UTC+0 的日期应为 2019-02-25，而不是 2019-02-26。
2. Timestamp 必须是当前系统时间，且需确保系统时间和标准时间是同步的，如果相差超过五分钟则必定失败。如果长时间不和标准时间同步，可能运行一段时间后，请求失败，返回签名过期错误。

根据以上规则，示例中得到的待签名字符串如下：

```
TC3-HMAC-SHA256
1551113065
2019-02-25/cvm/tc3_request
7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84
```

3. 计算签名

1）计算派生签名密钥，伪代码如下：

```
SecretKey = "********************************"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

派生出的密钥 `SecretDate`、`SecretService` 和 `SecretSigning` 是二进制的数据，可能包含不可打印字符，将其转为十六进制字符串打印的输出分别为：
da98fb70dcf6b112dc21038d1eeeb3a95c74b4dcb12c1131f864f6066bd02be0，
8d70cbefb03939f929db64d32dc2ba89b1095620119fe3e050e2b18c5bd2752f，
b596b923aad85185e2d1f6659d2a062e0a86731226e021e61bfe06f7ed05f5af。

请注意，不同的编程语言，HMAC 库函数中参数顺序可能不一样，请以实际情况为准。此处的伪代码密钥参数 key 在前，消息参数 data 在后。通常标准库函数会提供二进制格式的返回值，也可能会提供打印友好的十六进制格式的返回值，此处使用的是二进制格式。

字段名称	解释
SecretKey	原始的 SecretKey，即 `********************************`。
Date	即 Credential 中的 Date 字段信息。此示例取值为 `2019-02-25`。
Service	即 Credential 中的 Service 字段信息。此示例取值为 `cvm`。

2）计算签名，伪代码如下：

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

此示例计算结果是 `10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f`。

4. 拼接 Authorization

按如下格式拼接 Authorization：

```
Authorization =
Algorithm + ' ' +
'Credential=' + SecretId + '/' + CredentialScope + ', ' +
'SignedHeaders=' + SignedHeaders + ', ' +
'Signature=' + Signature
```

字段名称	解释
Algorithm	签名方法，固定为 `TC3-HMAC-SHA256`。
SecretId	密钥对中的 SecretId，即 `AKID********************************`。
CredentialScope	见上文，凭证范围。此示例计算结果是 `2019-02-25/cvm/tc3_request`。
SignedHeaders	见上文，参与签名的头部信息。此示例取值为 `content-type;host;x-tc-action`。
Signature	签名值。此示例计算结果是 `10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f`。

根据以上规则，示例中得到的值为：

```
TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;hos
t;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
```

最终完整的调用信息如下：

```
POST https://cvm.tencentcloudapi.com/
Authorization: TC3-HMAC-SHA256 Credential=AKID********************************/2019-02-25/cvm/tc3_request, SignedHeaders=c
ontent-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
Content-Type: application/json; charset=utf-8
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1551113065
X-TC-Region: ap-guangzhou

{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}
```

注意：

请求发送时的 HTTP 头部（Header）和请求体（Payload）必须和签名计算过程中的内容完全一致，否则会返回签名不一致错误。可以通过打印实际请求内容，网络抓包等方式对比排查。

签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 SDK 中心。当前支持的编程语言有：

- Python
- Java
- PHP
- Go
- NodeJS
- .NET
- C++
- Ruby

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- Signature Demo

为了更清楚地解释签名过程，下面以实际编程语言为例，将上述的签名过程完整实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

Java

```
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPITC3Demo {
private final static Charset UTF8 = StandardCharsets.UTF_8;
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
private final static String SECRET_ID = System.getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
```

```
private final static String SECRET_KEY = System.getenv("TENCENTCLOUD_SECRET_KEY");
private final static String CT_JSON = "application/json; charset=utf-8";

public static byte[] hmac256(byte[] key, String msg) throws Exception {
Mac mac = Mac.getInstance("HmacSHA256");
SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
mac.init(secretKeySpec);
return mac.doFinal(msg.getBytes(UTF8));
}

public static String sha256Hex(String s) throws Exception {
MessageDigest md = MessageDigest.getInstance("SHA-256");
byte[] d = md.digest(s.getBytes(UTF8));
return DatatypeConverter.printHexBinary(d).toLowerCase();
}

public static void main(String[] args) throws Exception {
String service = "cvm";
String host = "cvm.tencentcloudapi.com";
String region = "ap-guangzhou";
String action = "DescribeInstances";
String version = "2017-03-12";
String algorithm = "TC3-HMAC-SHA256";
String timestamp = "1551113065";
//String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
// 注意时区，否则容易出错
sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

// ************* 步骤 1: 拼接规范请求串 *************
String httpRequestMethod = "POST";
String canonicalUri = "/";
String canonicalQueryString = "";
String canonicalHeaders = "content-type:application/json; charset=utf-8\n"
+ "host:" + host + "\n" + "x-tc-action:" + action.toLowerCase() + "\n";
String signedHeaders = "content-type;host;x-tc-action";

String payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}";
String hashedRequestPayload = sha256Hex(payload);
String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryString + "\n"
+ canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
System.out.println(canonicalRequest);

// ************* 步骤 2: 拼接待签名字符串 *************
String credentialScope = date + "/" + service + "/" + "tc3_request";
String hashedCanonicalRequest = sha256Hex(canonicalRequest);
String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
System.out.println(stringToSign);

// ************* 步骤 3: 计算签名 *************
byte[] secretDate = hmac256(("TC3" + SECRET_KEY).getBytes(UTF8), date);
byte[] secretService = hmac256(secretDate, service);
byte[] secretSigning = hmac256(secretService, "tc3_request");
String signature = DatatypeConverter.printHexBinary(hmac256(secretSigning, stringToSign)).toLowerCase();
System.out.println(signature);
```

```
// ************* 步骤 4: 拼接 Authorization *************
String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
System.out.println(authorization);

TreeMap<String, String> headers = new TreeMap<String, String>();
headers.put("Authorization", authorization);
headers.put("Content-Type", CT_JSON);
headers.put("Host", host);
headers.put("X-TC-Action", action);
headers.put("X-TC-Timestamp", timestamp);
headers.put("X-TC-Version", version);
headers.put("X-TC-Region", region);

StringBuilder sb = new StringBuilder();
sb.append("curl -X POST https://").append(host)
.append(" -H \"Authorization: ").append(authorization).append("\"")
.append(" -H \"Content-Type: application/json; charset=utf-8\"")
.append(" -H \"Host: ").append(host).append("\"")
.append(" -H \"X-TC-Action: ").append(action).append("\"")
.append(" -H \"X-TC-Timestamp: ").append(timestamp).append("\"")
.append(" -H \"X-TC-Version: ").append(version).append("\"")
.append(" -H \"X-TC-Region: ").append(region).append("\"")
.append(" -d '").append(payload).append("'");
System.out.println(sb.toString());
}
}
```

Python

```
# -*- coding: utf-8 -*-
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

service = "cvm"
host = "cvm.tencentcloudapi.com"
endpoint = "https://" + host
region = "ap-guangzhou"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
#timestamp = int(time.time())
timestamp = 1551113065
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 1, "Filters": [{"Values": [u"未命名"], "Name": "instance-name"}]}

# ************* 步骤 1: 拼接规范请求串 *************
http_request_method = "POST"
canonical_uri = "/"
canonical_querystring = ""
```

```
ct = "application/json; charset=utf-8"
payload = json.dumps(params)
canonical_headers = "content-type:%s\nhost:%s\nx-tc-action:%s\n" % (ct, host, action.lower())
signed_headers = "content-type;host;x-tc-action"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
canonical_uri + "\n" +
canonical_querystring + "\n" +
canonical_headers + "\n" +
signed_headers + "\n" +
hashed_request_payload)
print(canonical_request)

# ************* 步骤 2: 拼接待签名字符串 *************
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
str(timestamp) + "\n" +
credential_scope + "\n" +
hashed_canonical_request)
print(string_to_sign)


# ************* 步骤 3: 计算签名 *************
# 计算签名摘要函数
def sign(key, msg):
return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ************* 步骤 4: 拼接 Authorization *************
authorization = (algorithm + " " +
"Credential=" + secret_id + "/" + credential_scope + ", " +
"SignedHeaders=" + signed_headers + ", " +
"Signature=" + signature)
print(authorization)

print('curl -X POST ' + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + host + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + str(timestamp) + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + "'")
```

Golang

```
package main

import (
"crypto/hmac"
```

```
"crypto/sha256"
"encoding/hex"
"fmt"
"os"
"strings"
"time"
)

func sha256hex(s string) string {
b := sha256.Sum256([]byte(s))
return hex.EncodeToString(b[:])
}

func hmacsha256(s, key string) string {
hashed := hmac.New(sha256.New, []byte(key))
hashed.Write([]byte(s))
return string(hashed.Sum(nil))
}

func main() {
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
host := "cvm.tencentcloudapi.com"
algorithm := "TC3-HMAC-SHA256"
service := "cvm"
version := "2017-03-12"
action := "DescribeInstances"
region := "ap-guangzhou"
//var timestamp int64 = time.Now().Unix()
var timestamp int64 = 1551113065

// step 1: build canonical request string
httpRequestMethod := "POST"
canonicalURI := "/"
canonicalQueryString := ""
canonicalHeaders := fmt.Sprintf("content-type:%s\nhost:%s\nx-tc-action:%s\n",
"application/json; charset=utf-8", host, strings.ToLower(action))
signedHeaders := "content-type;host;x-tc-action"
payload := `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`
hashedRequestPayload := sha256hex(payload)
canonicalRequest := fmt.Sprintf("%s\n%s\n%s\n%s\n%s\n%s",
httpRequestMethod,
canonicalURI,
canonicalQueryString,
canonicalHeaders,
signedHeaders,
hashedRequestPayload)
fmt.Println(canonicalRequest)

// step 2: build string to sign
date := time.Unix(timestamp, 0).UTC().Format("2006-01-02")
credentialScope := fmt.Sprintf("%s/%s/tc3_request", date, service)
hashedCanonicalRequest := sha256hex(canonicalRequest)
string2sign := fmt.Sprintf("%s\n%d\n%s\n%s",
algorithm,
```

```
timestamp,
credentialScope,
hashedCanonicalRequest)
fmt.Println(string2sign)

// step 3: sign string
secretDate := hmacsha256(date, "TC3"+secretKey)
secretService := hmacsha256(service, secretDate)
secretSigning := hmacsha256("tc3_request", secretService)
signature := hex.EncodeToString([]byte(hmacsha256(string2sign, secretSigning)))
fmt.Println(signature)

// step 4: build authorization
authorization := fmt.Sprintf("%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm,
secretId,
credentialScope,
signedHeaders,
signature)
fmt.Println(authorization)

curl := fmt.Sprintf(`curl -X POST https://%s\
-H "Authorization: %s"\
-H "Content-Type: application/json; charset=utf-8"\
-H "Host: %s" -H "X-TC-Action: %s"\
-H "X-TC-Timestamp: %d"\
-H "X-TC-Version: %s"\
-H "X-TC-Region: %s"\
-d '%s'`, host, authorization, host, action, timestamp, version, region, payload)
fmt.Println(curl)
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$host = "cvm.tencentcloudapi.com";
$service = "cvm";
$version = "2017-03-12";
$action = "DescribeInstances";
$region = "ap-guangzhou";
// $timestamp = time();
$timestamp = 1551113065;
$algorithm = "TC3-HMAC-SHA256";

// step 1: build canonical request string
$httpRequestMethod = "POST";
$canonicalUri = "/";
$canonicalQueryString = "";
$canonicalHeaders = implode("\n", [
"content-type:application/json; charset=utf-8",
"host:".$host,
"x-tc-action:".strtolower($action),
```

```
""
]);
$signedHeaders = implode(";", [
"content-type",
"host",
"x-tc-action",
]);
$payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}';
$hashedRequestPayload = hash("SHA256", $payload);
$canonicalRequest = $httpRequestMethod."\n"
.$canonicalUri."\n"
.$canonicalQueryString."\n"
.$canonicalHeaders."\n"
.$signedHeaders."\n"
.$hashedRequestPayload;
echo $canonicalRequest.PHP_EOL;

// step 2: build string to sign
$date = gmdate("Y-m-d", $timestamp);
$credentialScope = $date."/".$service."/tc3_request";
$hashedCanonicalRequest = hash("SHA256", $canonicalRequest);
$stringToSign = $algorithm."\n"
.$timestamp."\n"
.$credentialScope."\n"
.$hashedCanonicalRequest;
echo $stringToSign.PHP_EOL;

// step 3: sign string
$secretDate = hash_hmac("SHA256", $date, "TC3".$secretKey, true);
$secretService = hash_hmac("SHA256", $service, $secretDate, true);
$secretSigning = hash_hmac("SHA256", "tc3_request", $secretService, true);
$signature = hash_hmac("SHA256", $stringToSign, $secretSigning);
echo $signature.PHP_EOL;

// step 4: build authorization
$authorization = $algorithm
." Credential=".$secretId."/".$credentialScope
.", SignedHeaders=".$signedHeaders.", Signature=".$signature;
echo $authorization.PHP_EOL;

$curl = "curl -X POST https://".$host
.' -H "Authorization: '.$authorization.'"'
.' -H "Content-Type: application/json; charset=utf-8"'
.' -H "Host: '.$host.'"'
.' -H "X-TC-Action: '.$action.'"'
.' -H "X-TC-Timestamp: '.$timestamp.'"'
.' -H "X-TC-Version: '.$version.'"'
.' -H "X-TC-Region: '.$region.'"'
." -d '".$payload."'";
echo $curl.PHP_EOL;
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'digest'
```

```
require 'json'
require 'time'
require 'openssl'

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

service = 'cvm'
host = 'cvm.tencentcloudapi.com'
endpoint = 'https://' + host
region = 'ap-guangzhou'
action = 'DescribeInstances'
version = '2017-03-12'
algorithm = 'TC3-HMAC-SHA256'
# timestamp = Time.now.to_i
timestamp = 1551113065
date = Time.at(timestamp).utc.strftime('%Y-%m-%d')

# ************* 步骤 1：拼接规范请求串 *************
http_request_method = 'POST'
canonical_uri = '/'
canonical_querystring = ''
canonical_headers = "content-type:application/json; charset=utf-8\nhost:#{host}\nx-tc-action:#{action.downcase}\n"
signed_headers = 'content-type;host;x-tc-action'
# params = { 'Limit' => 1, 'Filters' => [{ 'Name' => 'instance-name', 'Values' => ['未命名'] }] }
# payload = JSON.generate(params, { 'ascii_only' => true, 'space' => ' ' })
# json will generate in random order, to get specified result in example, we hard-code it here.
payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
hashed_request_payload = Digest::SHA256.hexdigest(payload)
canonical_request = [
http_request_method,
canonical_uri,
canonical_querystring,
canonical_headers,
signed_headers,
hashed_request_payload,
].join("\n")

puts canonical_request

# ************* 步骤 2：拼接待签名字符串 *************
credential_scope = date + '/' + service + '/' + 'tc3_request'
hashed_request_payload = Digest::SHA256.hexdigest(canonical_request)
string_to_sign = [
algorithm,
timestamp.to_s,
credential_scope,
hashed_request_payload,
].join("\n")
puts string_to_sign

# ************* 步骤 3：计算签名 *************
digest = OpenSSL::Digest.new('sha256')
secret_date = OpenSSL::HMAC.digest(digest, 'TC3' + secret_key, date)
```

```
secret_service = OpenSSL::HMAC.digest(digest, secret_date, service)
secret_signing = OpenSSL::HMAC.digest(digest, secret_service, 'tc3_request')
signature = OpenSSL::HMAC.hexdigest(digest, secret_signing, string_to_sign)
puts signature

# ************* 步骤 4：拼接 Authorization *************
authorization = "#{algorithm} Credential=#{secret_id}/#{credential_scope}, SignedHeaders=#{signed_headers}, Signature=#{si
gnature}"
puts authorization

puts 'curl -X POST ' + endpoint \
+ ' -H "Authorization: ' + authorization + '"' \
+ ' -H "Content-Type: application/json; charset=utf-8"' \
+ ' -H "Host: ' + host + '"' \
+ ' -H "X-TC-Action: ' + action + '"' \
+ ' -H "X-TC-Timestamp: ' + timestamp.to_s + '"' \
+ ' -H "X-TC-Version: ' + version + '"' \
+ ' -H "X-TC-Region: ' + region + '"' \
+ " -d '" + payload + "'"
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Security.Cryptography;
using System.Text;

public class Application
{
public static string SHA256Hex(string s)
{
using (SHA256 algo = SHA256.Create())
{
byte[] hashbytes = algo.ComputeHash(Encoding.UTF8.GetBytes(s));
StringBuilder builder = new StringBuilder();
for (int i = 0; i < hashbytes.Length; ++i)
{
builder.Append(hashbytes[i].ToString("x2"));
}
return builder.ToString();
}
}

public static byte[] HmacSHA256(byte[] key, byte[] msg)
{
using (HMACSHA256 mac = new HMACSHA256(key))
{
return mac.ComputeHash(msg);
}
}

public static Dictionary<String, String> BuildHeaders(string secretid,
string secretkey, string service, string endpoint, string region,
string action, string version, DateTime date, string requestPayload)
{
string datestr = date.ToString("yyyy-MM-dd");
```

```
DateTime startTime = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc);
long requestTimestamp = (long)Math.Round((date - startTime).TotalMilliseconds, MidpointRounding.AwayFromZero) / 1000;
// ************* 步骤 1: 拼接规范请求串 *************
string algorithm = "TC3-HMAC-SHA256";
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string contentType = "application/json";
string canonicalHeaders = "content-type:" + contentType + "; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.ToLower() + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string hashedRequestPayload = SHA256Hex(requestPayload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
Console.WriteLine(canonicalRequest);

// ************* 步骤 2: 拼接待签名字符串 *************
string credentialScope = datestr + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = SHA256Hex(canonicalRequest);
string stringToSign = algorithm + "\n"
+ requestTimestamp.ToString() + "\n"
+ credentialScope + "\n"
+ hashedCanonicalRequest;
Console.WriteLine(stringToSign);

// ************* 步骤 3: 计算签名 *************
byte[] tc3SecretKey = Encoding.UTF8.GetBytes("TC3" + secretkey);
byte[] secretDate = HmacSHA256(tc3SecretKey, Encoding.UTF8.GetBytes(datestr));
byte[] secretService = HmacSHA256(secretDate, Encoding.UTF8.GetBytes(service));
byte[] secretSigning = HmacSHA256(secretService, Encoding.UTF8.GetBytes("tc3_request"));
byte[] signatureBytes = HmacSHA256(secretSigning, Encoding.UTF8.GetBytes(stringToSign));
string signature = BitConverter.ToString(signatureBytes).Replace("-", "").ToLower();
Console.WriteLine(signature);

// ************* 步骤 4: 拼接 Authorization *************
string authorization = algorithm + " "
+ "Credential=" + secretid + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", "
+ "Signature=" + signature;
Console.WriteLine(authorization);

Dictionary<string, string> headers = new Dictionary<string, string>();
headers.Add("Authorization", authorization);
headers.Add("Host", endpoint);
headers.Add("Content-Type", contentType + "; charset=utf-8");
headers.Add("X-TC-Timestamp", requestTimestamp.ToString());
headers.Add("X-TC-Version", version);
headers.Add("X-TC-Action", action);
headers.Add("X-TC-Region", region);
return headers;
}
public static void Main(string[] args)
```

```
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";

// 此处由于示例规范的原因，采用时间戳2019-02-26 00:44:25，此参数作为示例，如果在项目中，您应当使用：
// DateTime date = DateTime.UtcNow;
// 注意时区，建议此时间统一采用UTC时间戳，否则容易出错
DateTime date = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc).AddSeconds(1551113065);
string requestPayload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name
\"}]}";

Dictionary<string, string> headers = BuildHeaders(SECRET_ID, SECRET_KEY, service
, endpoint, region, action, version, date, requestPayload);

Console.WriteLine("POST https://cvm.tencentcloudapi.com");
foreach (KeyValuePair<string, string> kv in headers)
{
Console.WriteLine(kv.Key + ": " + kv.Value);
}
Console.WriteLine();
Console.WriteLine(requestPayload);
}
}
```

NodeJS

```
const crypto = require('crypto');

function sha256(message, secret = '', encoding) {
const hmac = crypto.createHmac('sha256', secret)
return hmac.update(message).digest(encoding)
}

function getHash(message, encoding = 'hex') {
const hash = crypto.createHash('sha256')
return hash.update(message).digest(encoding)
}

function getDate(timestamp) {
const date = new Date(timestamp * 1000)
const year = date.getUTCFullYear()
const month = ('0' + (date.getUTCMonth() + 1)).slice(-2)
const day = ('0' + date.getUTCDate()).slice(-2)
return `${year}-${month}-${day}`
}

function main(){
```

```
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const service = "cvm"
const region = "ap-guangzhou"
const action = "DescribeInstances"
const version = "2017-03-12"
//const timestamp = getTime()
const timestamp = 1551113065
//时间处理，获取世界时间日期
const date = getDate(timestamp)

// ************* 步骤 1：拼接规范请求串 *************
const payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}"

const hashedRequestPayload = getHash(payload);
const httpRequestMethod = "POST"
const canonicalUri = "/"
const canonicalQueryString = ""
const canonicalHeaders = "content-type:application/json; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.toLowerCase() + "\n"
const signedHeaders = "content-type;host;x-tc-action"

const canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload
console.log(canonicalRequest)

// ************* 步骤 2：拼接待签名字符串 *************
const algorithm = "TC3-HMAC-SHA256"
const hashedCanonicalRequest = getHash(canonicalRequest);
const credentialScope = date + "/" + service + "/" + "tc3_request"
const stringToSign = algorithm + "\n" +
timestamp + "\n" +
credentialScope + "\n" +
hashedCanonicalRequest
console.log(stringToSign)

// ************* 步骤 3：计算签名 *************
const kDate = sha256(date, 'TC3' + SECRET_KEY)
const kService = sha256(service, kDate)
const kSigning = sha256('tc3_request', kService)
const signature = sha256(stringToSign, kSigning, 'hex')
console.log(signature)

// ************* 步骤 4：拼接 Authorization *************
const authorization = algorithm + " " +
"Credential=" + SECRET_ID + "/" + credentialScope + ", " +
"SignedHeaders=" + signedHeaders + ", " +
```

```
"Signature=" + signature
console.log(authorization)

const curlcmd = 'curl -X POST ' + "https://" + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + endpoint + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + timestamp.toString() + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + "'"
console.log(curlcmd)
}
main()
```

C++

```
#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <iomanip>
#include <sstream>
#include <string>
#include <stdio.h>
#include <time.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

using namespace std;

string get_data(int64_t &timestamp)
{
string utcDate;
char buff[20] = {0};
// time_t timenow;
struct tm sttime;
sttime = *gmtime(&timestamp);
strftime(buff, sizeof(buff), "%Y-%m-%d", &sttime);
utcDate = string(buff);
return utcDate;
}

string int2str(int64_t n)
{
std::stringstream ss;
ss << n;
return ss.str();
}

string sha256Hex(const string &str)
{
char buf[3];
unsigned char hash[SHA256_DIGEST_LENGTH];
SHA256_CTX sha256;
SHA256_Init(&sha256);
```

```
SHA256_Update(&sha256, str.c_str(), str.size());
SHA256_Final(hash, &sha256);
std::string NewString = "";
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
snprintf(buf, sizeof(buf), "%02x", hash[i]);
NewString = NewString + buf;
}
return NewString;
}

string HmacSha256(const string &key, const string &input)
{
unsigned char hash[32];

HMAC_CTX *h;
#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX hmac;
HMAC_CTX_init(&hmac);
h = &hmac;
#else
h = HMAC_CTX_new();
#endif

HMAC_Init_ex(h, &key[0], key.length(), EVP_sha256(), NULL);
HMAC_Update(h, ( unsigned char* )&input[0], input.length());
unsigned int len = 32;
HMAC_Final(h, hash, &len);

#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX_cleanup(h);
#else
HMAC_CTX_free(h);
#endif

std::stringstream ss;
ss << std::setfill('0');
for (int i = 0; i < len; i++)
{
ss << hash[i];
}

return (ss.str());
}

string HexEncode(const string &input)
{
static const char* const lut = "0123456789abcdef";
size_t len = input.length();

string output;
output.reserve(2 * len);
for (size_t i = 0; i < len; ++i)
{
const unsigned char c = input[i];
output.push_back(lut[c >> 4]);
output.push_back(lut[c & 15]);
```

```
}
return output;
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string host = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";
int64_t timestamp = 1551113065;
string date = get_data(timestamp);

// ************* 步骤 1: 拼接规范请求串 *************
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string lower = action;
std::transform(action.begin(), action.end(), lower.begin(), ::tolower);
string canonicalHeaders = string("content-type:application/json; charset=utf-8\n")
+ "host:" + host + "\n"
+ "x-tc-action:" + lower + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}";
string hashedRequestPayload = sha256Hex(payload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
cout << canonicalRequest << endl;

// ************* 步骤 2: 拼接待签名字符串 *************
string algorithm = "TC3-HMAC-SHA256";
string RequestTimestamp = int2str(timestamp);
string credentialScope = date + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = sha256Hex(canonicalRequest);
string stringToSign = algorithm + "\n" + RequestTimestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
cout << stringToSign << endl;

// ************* 步骤 3: 计算签名 **************
string kKey = "TC3" + SECRET_KEY;
string kDate = HmacSha256(kKey, date);
string kService = HmacSha256(kDate, service);
string kSigning = HmacSha256(kService, "tc3_request");
string signature = HexEncode(HmacSha256(kSigning, stringToSign));
cout << signature << endl;

// ************* 步骤 4: 拼接 Authorization *************
```

```
string authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
cout << authorization << endl;

string curlcmd = "curl -X POST https://" + host + "\n"
+ " -H \"Authorization: " + authorization + "\"\n"
+ " -H \"Content-Type: application/json; charset=utf-8\"" + "\n"
+ " -H \"Host: " + host + "\"\n"
+ " -H \"X-TC-Action: " + action + "\"\n"
+ " -H \"X-TC-Timestamp: " + RequestTimestamp + "\"\n"
+ " -H \"X-TC-Version: " + version + "\"\n"
+ " -H \"X-TC-Region: " + region + "\"\n"
+ " -d '" + payload + "\'";
cout << curlcmd << endl;
return 0;
};
```

C

```
#include <ctype.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <stdint.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

void get_utc_date(int64_t timestamp, char* utc, int len)
{
// time_t timenow;
struct tm sttime;
sttime = *gmtime(&timestamp);
strftime(utc, len, "%Y-%m-%d", &sttime);
}

void sha256_hex(const char* str, char* result)
{
char buf[3];
unsigned char hash[SHA256_DIGEST_LENGTH];
SHA256_CTX sha256;
SHA256_Init(&sha256);
SHA256_Update(&sha256, str, strlen(str));
SHA256_Final(hash, &sha256);
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
snprintf(buf, sizeof(buf), "%02x", hash[i]);
strcat(result, buf);
}
}

void hmac_sha256(const char* key, int key_len,
const char* input, int input_len,
unsigned char* output, unsigned int* output_len)
{
HMAC_CTX *h;
```

```
#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX hmac;
HMAC_CTX_init(&hmac);
h = &hmac;
#else
h = HMAC_CTX_new();
#endif

HMAC_Init_ex(h, key, key_len, EVP_sha256(), NULL);
HMAC_Update(h, ( unsigned char* )input, input_len);
HMAC_Final(h, output, output_len);


#if OPENSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX_cleanup(h);
#else
HMAC_CTX_free(h);
#endif

}

void hex_encode(const char* input, int input_len, char* output)
{
static const char* const lut = "0123456789abcdef";

char add_out[128] = {0};
char temp[2] = {0};
for (size_t i = 0; i < input_len; ++i)
{
const unsigned char c = input[i];
temp[0] = lut[c >> 4];
strcat(add_out, temp);
temp[0] = lut[c & 15];
strcat(add_out, temp);
}
strncpy(output, add_out, 128);
}

void lowercase(const char * src, char * dst)
{
for (int i = 0; src[i]; i++)
{
dst[i] = tolower(src[i]);
}
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const char* SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const char* SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");
const char* service = "cvm";
const char* host = "cvm.tencentcloudapi.com";
const char* region = "ap-guangzhou";
const char* action = "DescribeInstances";
```

```
const char* version = "2017-03-12";
int64_t timestamp = 1551113065;
char date[20] = {0};
get_utc_date(timestamp, date, sizeof(date));

// ************* 步骤 1: 拼接规范请求串 *************
const char* http_request_method = "POST";
const char* canonical_uri = "/";
const char* canonical_query_string = "";
char canonical_headers[100] = {"content-type:application/json; charset=utf-8\nhost:"};
strcat(canonical_headers, host);
strcat(canonical_headers, "\nx-tc-action:");
char value[100] = {0};
lowercase(action, value);
strcat(canonical_headers, value);
strcat(canonical_headers, "\n");
const char* signed_headers = "content-type;host;x-tc-action";
const char* payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name
\"}]}";
char hashed_request_payload[100] = {0};
sha256_hex(payload, hashed_request_payload);

char canonical_request[256] = {0};
sprintf(canonical_request, "%s\n%s\n%s\n%s\n%s\n%s", http_request_method,
canonical_uri, canonical_query_string, canonical_headers,
signed_headers, hashed_request_payload);
printf("%s\n", canonical_request);


// ************* 步骤 2: 拼接待签名字符串 *************
const char* algorithm = "TC3-HMAC-SHA256";
char request_timestamp[16] = {0};
sprintf(request_timestamp, "%d", timestamp);
char credential_scope[64] = {0};
strcat(credential_scope, date);
sprintf(credential_scope, "%s/%s/tc3_request", date, service);
char hashed_canonical_request[100] = {0};
sha256_hex(canonical_request, hashed_canonical_request);
char string_to_sign[256] = {0};
sprintf(string_to_sign, "%s\n%s\n%s\n%s", algorithm, request_timestamp,
credential_scope, hashed_canonical_request);
printf("%s\n", string_to_sign);

// ************* 步骤 3: 计算签名 **************
char k_key[64] = {0};
sprintf(k_key, "%s%s", "TC3", SECRET_KEY);
unsigned char k_date[64] = {0};
unsigned int output_len = 0;
hmac_sha256(k_key, strlen(k_key), date, strlen(date), k_date, &output_len);
unsigned char k_service[64] = {0};
hmac_sha256(k_date, output_len, service, strlen(service), k_service, &output_len);
unsigned char k_signing[64] = {0};
hmac_sha256(k_service, output_len, "tc3_request", strlen("tc3_request"), k_signing, &output_len);
unsigned char k_hmac_sha_sign[64] = {0};
hmac_sha256(k_signing, output_len, string_to_sign, strlen(string_to_sign), k_hmac_sha_sign, &output_len);

char signature[128] = {0};
```

```
hex_encode(k_hmac_sha_sign, output_len, signature);
printf("%s\n", signature);

// ************* 步骤 4: 拼接 Authorization *************
char authorization[512] = {0};
sprintf(authorization, "%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm, SECRET_ID, credential_scope, signed_headers, signature);
printf("%s\n", authorization);

char curlcmd[10240] = {0};
sprintf(curlcmd, "curl -X POST https://%s\n \
-H \"Authorization: %s\"\n \
-H \"Content-Type: application/json; charset=utf-8\"\n \
-H \"Host: %s\"\n \
-H \"X-TC-Action: %s\"\n \
-H \"X-TC-Timestamp: %s\"\n \
-H \"X-TC-Version: %s\"\n \
-H \"X-TC-Region: %s\"\n \
-d \'%s\'",
host, authorization, host, action, request_timestamp, version, region, payload);
printf("%s\n", curlcmd);
return 0;
}
```

其他语言

- Lua: GitHub
- Swift: GitHub
- Dart: GitHub
- Shell(Bash): GitHub

签名失败

存在以下签名失败的错误码，请根据实际情况处理。

错误码	错误描述
AuthFailure.SignatureExpire	签名过期。Timestamp 与服务器接收到请求的时间相差不得超过五分钟。
AuthFailure.SecretIdNotFound	密钥不存在。请到控制台查看密钥是否被禁用，是否少复制了字符或者多了字符。
AuthFailure.SignatureFailure	签名错误。可能是签名计算错误，或者签名与实际发送的内容不相符合，也有可能是密钥 SecretKey 错误导致的。
AuthFailure.TokenFailure	临时证书 Token 错误。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。

签名方法

最近更新时间：2024-12-25 01:31:35

签名方法 v1 简单易用，但是功能和安全性都不如签名方法 v3，推荐使用签名方法 v3。

首次接触，建议使用 API Explorer 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v1”，可以生成签名过程进行验证，并提供了部分编程语言的签名示例，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 Python、Java、PHP、Go、NodeJS、.NET、C++、Ruby。

推荐使用 API Explorer　　点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（Signature）以验证请求者身份。
签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往 云API密钥页面 申请，否则无法调用云 API 接口。

1. 申请安全凭证

在第一次使用云 API 之前，请前往 云 API 密钥页面 申请安全凭证。
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- **用户必须严格保管安全凭证，避免泄露。**

申请安全凭证的具体步骤如下：

1. 登录 腾讯云管理中心控制台 。
2. 前往 云 API 密钥 的控制台页面
3. 在 云 API 密钥 页面，单击【新建密钥】即可以创建一对 SecretId/SecretKey。

注意：每个账号最多可以拥有两对 SecretId/SecretKey。

2. 生成签名串

有了安全凭证 SecretId 和 SecretKey 后，就可以生成签名串了。以下是使用签名方法 v1 生成签名串的详细过程：

假设用户的 SecretId 和 SecretKey 分别是：

- SecretId: `AKID********************************`
- SecretKey: `********************************`

注意：这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作！

以云服务器查看实例列表（DescribeInstances）请求为例，当用户调用这一接口时，其请求参数可能如下：

参数名称	中文	参数值
Action	方法名	DescribeInstances
SecretId	密钥 ID	`AKID********************************`
Timestamp	当前时间戳	1465185768
Nonce	随机正整数	11886
Region	实例所在区域	ap-guangzhou
InstanceIds.0	待查询的实例 ID	ins-09dx96dg
Offset	偏移量	0
Limit	最大允许输出	20
Version	接口版本号	2017-03-12

这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数。

2.1. 对参数排序

首先对所有请求参数按参数名的字典序（ ASCII 码）升序排序。注意：1）只按参数名进行排序，参数值保持对应即可，不参与比大小；2）按 ASCII 码比大小，如 InstanceIds.2 要排在 InstanceIds.12 后面，不是按字母表，也不是按数值。用户可以借助编程语言中的相关排序函数来实现这一功能，如 PHP 中的 ksort 函数。上述示例参数的排序结果如下：

```
{
'Action' : 'DescribeInstances',
'InstanceIds.0' : 'ins-09dx96dg',
'Limit' : 20,
'Nonce' : 11886,
'Offset' : 0,
'Region' : 'ap-guangzhou',
'SecretId' : 'AKID********************************',
'Timestamp' : 1465185768,
'Version': '2017-03-12',
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

2.2. 拼接请求字符串

此步骤生成请求字符串。
将把上一步排序好的请求参数格式化成“参数名称=参数值”的形式，如对 Action 参数，其参数名称为 "Action" ，参数值为 "DescribeInstances" ，因此格式化后就为 Action=DescribeInstances 。
注意：“参数值”为原始值而非 url 编码后的值。

然后将格式化后的各个参数用"&"拼接在一起，最终生成的请求字符串为：

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*******
************************&Timestamp=1465185768&Version=2017-03-12
```

2.3. 拼接签名原文字符串

此步骤生成签名原文字符串。
签名原文字符串由以下几个参数构成：

1. 请求方法: 支持 POST 和 GET 方式，这里使用 GET 请求，注意方法为全大写。
2. 请求主机:查看实例列表(DescribeInstances)的请求域名为：cvm.tencentcloudapi.com。实际的请求域名根据接口所属模块的不同而不同，详见各接口说明。
3. 请求路径: 当前版本云API的请求路径固定为 / 。
4. 请求字符串: 即上一步生成的请求字符串。

签名原文串的拼接规则为： 请求方法 + 请求主机 +请求路径 + ? + 请求字符串 。

示例的拼接结果为：

```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-gu
angzhou&SecretId=AKID******************************&Timestamp=1465185768&Version=2017-03-12
```

2.4. 生成签名串

此步骤生成签名串。
首先使用 HMAC-SHA1 算法对上一步中获得的**签名原文字符串**进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。

具体代码如下，以 PHP 语言为例：

```
$secretKey = '********************************';
$srcStr = 'GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&R
egion=ap-guangzhou&SecretId=AKID******************************&Timestamp=1465185768&Version=2017-03-12';
```

```
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

最终得到的签名串为：

```
7RAM2xfNMO9EiVTNmPg06MRnCvQ=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

3. 签名串编码

生成的签名串并不能直接作为请求参数，需要对其进行 URL 编码。

如上一步生成的签名串为 7RAM2xfNMO9EiVTNmPg06MRnCvQ= ，最终得到的签名串请求参数（Signature）为：7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D，它将用于生成最终的请求 URL。

注意：如果用户的请求方法是 GET，或者请求方法为 POST 同时 Content-Type 为 application/x-www-form-urlencoded，则发送请求时所有请求参数的值均需要做 URL 编码，参数键和=符号不需要编码。非 ASCII 字符在 URL 编码前需要先以 UTF-8 进行编码。

注意：有些编程语言的网络库会自动为所有参数进行 urlencode，在这种情况下，就不需要对签名串进行 URL 编码了，否则两次 URL 编码会导致签名失败。

注意：其他参数值也需要进行编码，编码采用 RFC 3986。使用 %XY 对特殊字符例如汉字进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F），使用小写将引发错误。

4. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理。

错误代码	错误描述
AuthFailure.SignatureExpire	签名过期
AuthFailure.SecretIdNotFound	密钥不存在
AuthFailure.SignatureFailure	签名错误
AuthFailure.TokenFailure	token 错误
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）

5. 签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0 ，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 SDK 中心。当前支持的编程语言有：

- Python
- Java
- PHP
- Go
- NodeJS
- .NET
- C++
- Ruby

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- Signature Demo

为了更清楚的解释签名过程，下面以实际编程语言为例，将上述的签名过程具体实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK 。

最终输出的 url 可能为： `https://cvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-`

guangzhou&SecretId=AKID********************************&Signature=7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D&Timestamp=1465185768&Version=2017-03-12 。

注意：由于示例中的密钥是虚构的，时间戳也不是系统当前时间，因此如果将此 url 在浏览器中打开或者用 curl 等命令调用时会返回鉴权错误：签名过期。为了得到一个可以正常返回的 url ，需要修改示例中的 SecretId 和 SecretKey 为真实的密钥，并使用系统当前时间戳作为 Timestamp 。

注意：在下面的示例中，不同编程语言，甚至同一语言每次执行得到的 url 可能都有所不同，表现为参数的顺序不同，但这并不影响正确性。只要所有参数都在，且签名计算正确即可。

注意：以下代码仅适用于 API 3.0，不能直接用于其他的签名流程，请以对应的实际文档为准。

Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPIDemo {
private final static String CHARSET = "UTF-8";

public static String sign(String s, String key, String method) throws Exception {
Mac mac = Mac.getInstance(method);
SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
mac.init(secretKeySpec);
byte[] hash = mac.doFinal(s.getBytes(CHARSET));
return DatatypeConverter.printBase64Binary(hash);
}

public static String getStringToSign(TreeMap<String, Object> params) {
StringBuilder s2s = new StringBuilder("GETcvm.tencentcloudapi.com/?");
// 签名时要求对参数进行字典排序，此处用TreeMap保证顺序
for (String k : params.keySet()) {
s2s.append(k).append("=").append(params.get(k).toString()).append("&");
}
return s2s.toString().substring(0, s2s.length() - 1);
}

public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException {
StringBuilder url = new StringBuilder("https://cvm.tencentcloudapi.com/?");
// 实际请求的url中对参数顺序没有要求
for (String k : params.keySet()) {
// 需要对请求串进行urlencode，由于key都是英文字母，故此处仅对其value进行urlencode
url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).append("&");
}
return url.toString().substring(0, url.length() - 1);
}

public static void main(String[] args) throws Exception {
TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap可以自动排序
// 实际调用时应当使用随机数，例如：params.put("Nonce", new Random().nextInt(java.lang.Integer.MAX_VALUE));
params.put("Nonce", 11886); // 公共参数
// 实际调用时应当使用系统当前时间，例如： params.put("Timestamp", System.currentTimeMillis() / 1000);
params.put("Timestamp", 1465185768); // 公共参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
params.put("SecretId", System.getenv("TENCENTCLOUD_SECRET_ID")); // 公共参数
```

```
params.put("Action", "DescribeInstances"); // 公共参数
params.put("Version", "2017-03-12"); // 公共参数
params.put("Region", "ap-guangzhou"); // 公共参数
params.put("Limit", 20); // 业务参数
params.put("Offset", 0); // 业务参数
params.put("InstanceIds.0", "ins-09dx96dg"); // 业务参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
params.put("Signature", sign(getStringToSign(params), System.getenv("TENCENTCLOUD_SECRET_KEY"), "HmacSHA1")); // 公共参数
System.out.println(getUrl(params));
}
}
```

Python

注意：如果是在 Python 2 环境中运行，需要先安装 requests 依赖包： `pip install requests` 。

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import os
import time

import requests

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

def get_string_to_sign(method, endpoint, params):
s = method + endpoint + "/?"
query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
return s + query_str

def sign_str(key, s, method):
hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
return base64.b64encode(hmac_str)

if __name__ == '__main__':
endpoint = "cvm.tencentcloudapi.com"
data = {
'Action' : 'DescribeInstances',
'InstanceIds.0' : 'ins-09dx96dg',
'Limit' : 20,
'Nonce' : 11886,
'Offset' : 0,
'Region' : 'ap-guangzhou',
'SecretId' : secret_id,
'Timestamp' : 1465185768, # int(time.time())
'Version': '2017-03-12'
}
s = get_string_to_sign("GET", endpoint, data)
data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
print(data["Signature"])
# 此处会实际调用，成功后可能产生计费
```

```
# resp = requests.get("https://" + endpoint, params=data)
# print(resp.url)
```

Golang

```
package main

import (
"bytes"
"crypto/hmac"
"crypto/sha1"
"encoding/base64"
"fmt"
"os"
"sort"
"strconv"
)

func main() {
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID******************************
secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ******************************
secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
params := map[string]string{
"Nonce": "11886",
"Timestamp": strconv.Itoa(1465185768),
"Region": "ap-guangzhou",
"SecretId": secretId,
"Version": "2017-03-12",
"Action": "DescribeInstances",
"InstanceIds.0": "ins-09dx96dg",
"Limit": strconv.Itoa(20),
"Offset": strconv.Itoa(0),
}

var buf bytes.Buffer
buf.WriteString("GET")
buf.WriteString("cvm.tencentcloudapi.com")
buf.WriteString("/")
buf.WriteString("?")

// sort keys by ascii asc order
keys := make([]string, 0, len(params))
for k, _ := range params {
keys = append(keys, k)
}
sort.Strings(keys)

for i := range keys {
k := keys[i]
buf.WriteString(k)
buf.WriteString("=")
buf.WriteString(params[k])
buf.WriteString("&")
}
buf.Truncate(buf.Len() - 1)
```

```
hashed := hmac.New(sha1.New, []byte(secretKey))
hashed.Write(buf.Bytes())

fmt.Println(base64.StdEncoding.EncodeToString(hashed.Sum(nil)))
}
```

PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$param["Nonce"] = 11886;//rand();
$param["Timestamp"] = 1465185768;//time();
$param["Region"] = "ap-guangzhou";
$param["SecretId"] = $secretId;
$param["Version"] = "2017-03-12";
$param["Action"] = "DescribeInstances";
$param["InstanceIds.0"] = "ins-09dx96dg";
$param["Limit"] = 20;
$param["Offset"] = 0;

ksort($param);

$signStr = "GETcvm.tencentcloudapi.com/?";
foreach ( $param as $key => $value ) {
$signStr = $signStr . $key . "=" . $value . "&";
}
$signStr = substr($signStr, 0, -1);

$signature = base64_encode(hash_hmac("sha1", $signStr, $secretKey, true));
echo $signature.PHP_EOL;
// need to install and enable curl extension in php.ini
// $param["Signature"] = $signature;
// $url = "https://cvm.tencentcloudapi.com/?".http_build_query($param);
// echo $url.PHP_EOL;
// $ch = curl_init();
// curl_setopt($ch, CURLOPT_URL, $url);
// $output = curl_exec($ch);
// curl_close($ch);
// echo json_decode($output);
```

Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'time'
require 'openssl'
require 'base64'

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
```

```
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

method = 'GET'
endpoint = 'cvm.tencentcloudapi.com'
data = {
'Action' => 'DescribeInstances',
'InstanceIds.0' => 'ins-09dx96dg',
'Limit' => 20,
'Nonce' => 11886,
'Offset' => 0,
'Region' => 'ap-guangzhou',
'SecretId' => secret_id,
'Timestamp' => 1465185768, # Time.now.to_i
'Version' => '2017-03-12',
}
sign = method + endpoint + '/?'
params = []
data.sort.each do |item|
params << "#{item[0]}=#{item[1]}"
end
sign += params.join('&')
digest = OpenSSL::Digest.new('sha1')
data['Signature'] = Base64.encode64(OpenSSL::HMAC.digest(digest, secret_key, sign))
puts data['Signature']

# require 'net/http'
# uri = URI('https://' + endpoint)
# uri.query = URI.encode_www_form(data)
# p uri
# res = Net::HTTP.get_response(uri)
# puts res.body
```

DotNet

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Security.Cryptography;
using System.Text;

public class Application {
public static string Sign(string signKey, string secret)
{
string signRet = string.Empty;
using (HMACSHA1 mac = new HMACSHA1(Encoding.UTF8.GetBytes(signKey)))
{
byte[] hash = mac.ComputeHash(Encoding.UTF8.GetBytes(secret));
signRet = Convert.ToBase64String(hash);
}
return signRet;
}
public static string MakeSignPlainText(SortedDictionary<string, string> requestParams, string requestMethod, string reques
tHost, string requestPath)
{
string retStr = "";
retStr += requestMethod;
```

```
retStr += requestHost;
retStr += requestPath;
retStr += "?";
string v = "";
foreach (string key in requestParams.Keys)
{
v += string.Format("{0}={1}&", key, requestParams[key]);
}
retStr += v.TrimEnd('&');
return retStr;
}

public static void Main(string[] args)
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";
double RequestTimestamp = 1465185768; // 时间戳 2019-02-26 00:44:25,此参数作为示例，以实际为准
// long timestamp = ToTimestamp() / 1000;
// string requestTimestamp = timestamp.ToString();
Dictionary<string, string> param = new Dictionary<string, string>();
param.Add("Limit", "20");
param.Add("Offset", "0");
param.Add("InstanceIds.0", "ins-09dx96dg");
param.Add("Action", action);
param.Add("Nonce", "11886");
// param.Add("Nonce", Math.Abs(new Random().Next()).ToString());

param.Add("Timestamp", RequestTimestamp.ToString());
param.Add("Version", version);

param.Add("SecretId", SECRET_ID);
param.Add("Region", region);
SortedDictionary<string, string> headers = new SortedDictionary<string, string>(param, StringComparer.Ordinal);
string sigInParam = MakeSignPlainText(headers, "GET", endpoint, "/");
string sigOutParam = Sign(SECRET_KEY, sigInParam);
Console.WriteLine(sigOutParam);
}
}
```

NodeJS

```
const crypto = require('crypto');

function get_req_url(params, endpoint){
params['Signature'] = encodeURIComponent(params['Signature']);
const url_strParam = sort_params(params)
return "https://" + endpoint + "/?" + url_strParam.slice(1);
}
```

```
function formatSignString(reqMethod, endpoint, path, strParam){
let strSign = reqMethod + endpoint + path + "?" + strParam.slice(1);
return strSign;
}
function sha1(secretKey, strsign){
let signMethodMap = {'HmacSHA1': "sha1"};
let hmac = crypto.createHmac(signMethodMap['HmacSHA1'], secretKey || "");
return hmac.update(Buffer.from(strsign, 'utf8')).digest('base64')
}

function sort_params(params){
let strParam = "";
let keys = Object.keys(params);
keys.sort();
for (let k in keys) {
//k = k.replace(/_/g, '.');
strParam += ("&" + keys[k] + "=" + params[keys[k]]);
}
return strParam
}

function main(){
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID********************************
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY，值为示例的 ********************************
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const Region = "ap-guangzhou"
const Version = "2017-03-12"
const Action = "DescribeInstances"
const Timestamp = 1465185768 // 时间戳 2016-06-06 12:02:48，此参数作为示例，以实际为准
// const Timestamp = Math.round(Date.now() / 1000)
const Nonce = 11886 // 随机正整数
//const nonce = Math.round(Math.random() * 65535)

let params = {};
params['Action'] = Action;
params['InstanceIds.0'] = 'ins-09dx96dg';
params['Limit'] = 20;
params['Offset'] = 0;
params['Nonce'] = Nonce;
params['Region'] = Region;
params['SecretId'] = SECRET_ID;
params['Timestamp'] = Timestamp;
params['Version'] = Version;

// 1. 对参数排序，并拼接请求字符串
strParam = sort_params(params)

// 2. 拼接签名原文字符串
const reqMethod = "GET";
const path = "/";
strSign = formatSignString(reqMethod, endpoint, path, strParam)
// console.log(strSign)
```

```
// 3. 生成签名串
params['Signature'] = sha1(SECRET_KEY, strSign)
console.log(params['Signature'])

// 4. 进行url编码并拼接请求url
// const req_url = get_req_url(params, endpoint)
// console.log(params['Signature'])
// console.log(req_url)
}
main()
```

返回结果

最近更新时间：2024-03-12 19:51:13

云 API 3.0 接口默认返回 JSON 数据，返回非 JSON 格式的接口会在文档中做出说明。返回 JSON 数据时最大限制为 50 MB，如果返回的数据超过最大限制，请求会失败并返回内部错误。请根据接口文档中给出的过滤功能（例如时间范围）或者分页功能，控制返回数据不要过大。

注意：目前只要请求被服务端正常处理了，响应的 HTTP 状态码均为200。例如返回的消息体里的错误码是签名失败，但 HTTP 状态码是200，而不是401。

正确返回结果

以云服务器的接口查看实例状态列表 (DescribeInstancesStatus) 2017-03-12 版本为例，若调用成功，其可能的返回如下为：

```
{
"Response": {
"TotalCount": 0,
"InstanceStatusSet": [],
"RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
}
}
```

- Response 及其内部的 RequestId 是固定的字段，无论请求成功与否，只要 API 处理了，则必定会返回。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 腾讯云客服 或 提交工单，并提供该 ID 来解决问题。
- 除了固定的字段外，其余均为具体接口定义的字段，不同的接口所返回的字段参见接口文档中的定义。此例中的 TotalCount 和 InstanceStatusSet 均为 DescribeInstancesStatus 接口定义的字段，由于调用请求的用户暂时还没有云服务器实例，因此 TotalCount 在此情况下的返回值为 0， InstanceStatusSet 列表为空。

错误返回结果

若调用失败，其返回值示例如下为：

```
{
"Response": {
"Error": {
"Code": "AuthFailure.SignatureFailure",
"Message": "The provided credentials could not be validated. Please check your signature is correct."
},
"RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
}
}
```

- Error 的出现代表着该请求调用失败。Error 字段连同其内部的 Code 和 Message 字段在调用失败时是必定返回的。
- Code 表示具体出错的错误码，当请求出错时可以先根据该错误码在公共错误码和当前接口对应的错误码列表里面查找对应原因和解决方案。
- Message 显示出了这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新，用户不应依赖这个返回值。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 腾讯云客服 或 提交工单，并提供该 ID 来解决问题。

公共错误码

返回结果中如果存在 Error 字段，则表示调用 API 接口失败。 Error 中的 Code 字段表示错误码，所有业务都可能出现的错误码为公共错误码。完整的错误码列表请参考本产品“API 文档”目录下的“错误码”页面。

参数类型

最近更新时间：2022-08-10 06:29:23

目前腾讯云 API 3.0 输入参数和输出参数支持如下几种数据格式：

- String: 字符串。
- Integer：整型，上限为无符号64位整数。SDK 3.0 不同编程语言支持的类型有所差异，建议以所使用编程语言的最大整型定义，例如 `Golang` 的 `uint64`。
- Boolean：布尔型。
- Float：浮点型。
- Double：双精度浮点型。
- Date：字符串，日期格式。例如：2022-01-01。
- Timestamp：字符串，时间格式。例如：2022-01-01 00:00:00。
- Timestamp ISO8601：ISO 8601 是由国际标准化组织（International Organization for Standardization，ISO）发布的关于日期和时间格式的国际标准，对应国标《GB/T 7408-2005数据元和交换格式信息交换日期和时间表示法》。建议以所使用编程语言的标准库进行格式解析。例如：2022-01-01T00:00:00+08:00。
- Binary：二进制内容，需要以特定协议请求和解析。

数据传输服务相关接口
查询数据迁移任务

最近更新时间：2025-04-25 01:29:05

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

查询数据迁移任务.
如果是金融区链路, 请使用域名: https://dts.ap-shenzhen-fsi.tencentcloudapi.com

默认接口请求频率限制：100次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrateJobs。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	否	String	数据迁移任务ID 示例值：dts-83i2usk2
JobName	否	String	数据迁移任务名称 示例值：dts-job1
Order	否	String	排序字段，可以取值为JobId、Status、JobName、MigrateType、RunMode、CreateTime 示例值：CreateTime
OrderSeq	否	String	排序方式，升序为ASC，降序为DESC 示例值：DESC
Offset	否	Integer	偏移量，默认为0 示例值：0
Limit	否	Integer	返回实例数量，默认20，有效区间[1,100] 示例值：2
TagFilters.N	否	Array of TagFilter	标签过滤条件

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	任务数目 示例值：2
JobList	Array of MigrateJobInfo	任务详情数组
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询数据迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeMigrateJobs
&Order=CreateTime
&OrderSeq=DESC
&Offset=0
&Limit=2
&<公共请求参数>
```

输出示例

```
{
"Response": {
"TotalCount": 1,
"JobList": [
{
"JobId": "dts-1kl0iy0v",
"JobName": "userdts",
"MigrateOption": {
"RunMode": 1,
"ExpectTime": "0000-00-00 00:00:00",
"MigrateType": 2,
"MigrateObject": 2,
"ConsistencyType": 5,
"IsOverrideRoot": 0,
"ExternParams": "",
"ConsistencyParams": {
"SelectRowsPerTable": 0,
"TablesSelectAll": 0,
"TablesSelectCount": 0
}
},
"SrcAccessType": "extranet",
"SrcDatabaseType": "mysql",
"SrcInfo": {
"AccessKey": "",
"Ip": "9.18.84.24",
"Port": 10304,
"User": "root",
"Password": "",
"RdsInstanceId": "",
"CvmInstanceId": "",
"UniqDcgId": "",
"SubnetId": "",
"UniqVpnGwId": "",
"InstanceId": "",
"Region": "ap-guangzhou",
"VpcId": "",
"CcnId": "",
"EngineVersion": "",
"Supplier": ""
},
"DstAccessType": "cdb",
"DstDatabaseType": "mysql",
```

```
"DstInfo": {
"InstanceId": "cdb-l78e0nbv",
"Ip": "",
"Port": 0,
"Region": "ap-shanghai",
"ReadOnly": 0,
"User": "",
"Password": ""
},
"Tags": [
{
"TagKey": "负责人",
"TagValue": "alice"
}
],
"Detail": {
"StepAll": 0,
"StepNow": 0,
"Progress": "0",
"CurrentStepProgress": "0",
"MasterSlaveDistance": 0,
"SecondsBehindMaster": 0,
"StepInfo": []
},
"Status": 1,
"DatabaseInfo": "",
"CreateTime": "2018-05-24 15:06:03",
"StartTime": "0000-00-00 00:00:00",
"EndTime": "0000-00-00 00:00:00",
"ErrorInfo": [],
"SrcInfoMulti": []
}
],
"RequestId": "c032aab5-b56a-428d-9cf7-e5f324ee408b"
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

版权所有：腾讯云计算（北京）有限责任公司

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.InternalErrorError	内部错误。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.BizInvalidParameterValueError	业务参数错误。
InvalidParameterValue.DataConvertError	数据转换错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

撤销数据迁移任务

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（StopMigrateJob）用于撤销数据迁移任务。
在迁移过程中允许调用该接口撤销迁移，撤销迁移的任务会失败。通过DescribeMigrateJobs接口查询到任务状态为运行中（status=7）或准备完成（status=8）时，才能撤销数据迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StopMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 撤销数据迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=StopMigrateJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "22f6b9ca-e94d-4a4d-b48f-a5133e791374"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
OperationDenied.JobOperationDeniedError	任务操作失败。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

启动数据迁移任务

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（StartMigrationJob）用于启动迁移任务。非定时迁移任务会在调用后立即开始迁移，定时任务则会开始倒计时。
调用此接口前，请务必先使用CreateMigrateCheckJob校验数据迁移任务，并通过DescribeMigrateJobs接口查询到任务状态为校验通过（status=4）时，才能启动数据迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：StartMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 启动数据迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=StartMigrateJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "bab78b9b-ce8a-4645-bfb2-5b03397d6ea0"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StartJobFailed	任务开始失败。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.AddTaskError	添加异步任务失败。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.LockError	锁冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

修改数据迁移任务

最近更新时间：2025-04-25 01:29:05

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（ModifyMigrateJob）用于修改数据迁移任务。
当迁移任务处于下述状态时，允许调用本接口修改迁移任务：迁移创建中（status=1）、 校验成功(status=4)、校验失败(status=5)、迁移失败(status=10)。但源实例、目标实例类型和目标实例地域不允许修改。

如果是金融区链路, 请使用域名: dts.ap-shenzhen-fsi.tencentcloudapi.com

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifyMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	待修改的数据迁移任务ID 示例值：dts-75sekyal
JobName	否	String	数据迁移任务名称 示例值：dts-75sekyal
MigrateOption	否	MigrateOption	迁移任务配置选项
SrcAccessType	否	String	源实例接入类型，值包括：extranet(外网),cvm(CVM自建实例),dcg(专线接入的实例),vpncloud(云VPN接入的实例),cdb(云上CDB实例) 示例值：cvm
SrcInfo	否	SrcInfo	源实例信息，具体内容跟迁移任务类型相关
DstAccessType	否	String	目标实例接入类型，值包括：extranet(外网),cvm(CVM自建实例),dcg(专线接入的实例),vpncloud(云VPN接入的实例)，cdb(云上CDB实例). 目前只支持cdb. 示例值：cdb
DstInfo	否	DstInfo	目标实例信息, 其中目标实例地域不允许修改.
DatabaseInfo	否	String	当选择'指定库表'迁移的时候, 需要设置待迁移的源数据库表信息,用符合json数组格式的字符串描述, 如下所例。 对于database-table两级结构的数据库： [{"Database":"db1","Table":["table1","table2"]},{"Database":"db2"}] 对于database-schema-table三级结构： [{"Database":"db1","Schema":"s1","Table":["table1","table2"]}, {"Database":"db1","Schema":"s2","Table":["table1","table2"]}, {"Database":"db2","Schema":"s1","Table":["table1","table2"]},{"Database":"db3"}, {"Database":"db4","Schema":"s1"}] 如果是'整个实例'的迁移模式,不需设置该字段 示例值：[{"Database":"db1","Table":["table1","table2"]},{"Database":"db2"}]
SrcNodeType	否	String	源实例类型: ""或者"simple":主从节点，"cluster": 集群节点 示例值：cluster

参数名称	必选	类型	描述
SrcInfoMulti.N	否	Array of SrcInfo	源实例信息，具体内容跟迁移任务类型相关

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据迁移任务

修改dts-1kl0iy0v迁移任务配置

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifyMigrateJob
&JobId=dts-1kl0iy0v
&JobName=userdts
&DatabaseInfo=[{"Database":"test","Table":["user","log"]}]
&MigrateOption.RunMode=1
&MigrateOption.MigrateType=2
&MigrateOption.MigrateObject=2
&MigrateOption.ConsistencyType=5
&MigrateOption.IsOverrideRoot=0
&DstInfo.ReadOnly=0
&DstInfo.Region=ap-guangzhou
&SrcInfo.Supplier=others
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "27ef2b7c-a786-48b4-9404-2f9baf3f4916"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee

版权所有：腾讯云计算（北京）有限责任公司

- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameterValue.BizInvalidParameterValueError	业务参数错误。
InvalidParameterValue.DataConvertError	数据转换错误。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

获取迁移校验结果

最近更新时间：2025-04-25 01:29:05

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于创建校验后,获取校验的结果. 能查询到当前校验的状态和进度.
若通过校验, 则可调用'StartMigrateJob' 开始迁移.
若未通过校验, 则能查询到校验失败的原因. 请按照报错, 通过'ModifyMigrateJob'修改迁移配置或是调整源/目标实例的相关参数.

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeMigrateCheckJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
Status	String	校验任务状态：unavailable(当前不可用), starting(开始中)，running(校验中)，finished(校验完成) 示例值：finished
ErrorCode	Integer	任务的错误码 示例值：-1
ErrorMessage	String	任务的错误信息 示例值：选择的库表不存在，请重新选择[存在输入参数table在源实例没有找到]
Progress	String	Check任务总进度,如："30"表示30% 示例值：100
CheckFlag	Integer	校验是否通过,0-未通过，1-校验通过, 3-未校验 示例值：0
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取迁移校验结果

校验成功

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeMigrateCheckJob
&JobId=dts-dau5czmd
```

```
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Status": "finished",
"ErrorCode": 0,
"ErrorMessage": "ok",
"Progress": "100",
"CheckFlag": 1,
"RequestId": "336448b0-2a45-4be4-8356-c245eab5784f"
}
}
```

示例2 获取迁移校验结果-2

校验失败

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeMigrateCheckJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Status": "finished",
"ErrorCode": -1,
"ErrorMessage": "选择的库表不存在，请重新选择[存在输入参数table在源实例没有找到]",
"Progress": "100",
"CheckFlag": 0,
"RequestId": "67b4cfcf-6957-48ae-b7ef-ba33209895e3"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee

版权所有：腾讯云计算（北京）有限责任公司

- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.ProtocolError	通信协议错误。
InvalidParameter	参数错误。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

删除数据迁移任务

最近更新时间：2025-04-25 01:29:05

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（DeleteMigrationJob）用于删除数据迁移任务。当通过DescribeMigrateJobs接口查询到任务的状态为：检验中（status=3）、运行中（status=7）、准备完成（status=8）、撤销中（status=11）或者完成中（status=12）时，不允许删除任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DeleteMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 删除数据迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=DeleteMigrateJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "e15f9b4c-9841-40d2-a28b-dea284a17315"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError	内部错误。
InternalError.CeleryError	内部调度系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
OperationDenied	操作被拒绝。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.MigrateServiceSupportError	数据迁移服务不支持当前迁移类型。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。

创建数据迁移任务

最近更新时间：2025-04-25 01:29:05

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（CreateMigrateJob）用于创建数据迁移任务。

如果是金融区链路, 请使用域名: dts.ap-shenzhen-fsi.tencentcloudapi.com

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobName	是	String	数据迁移任务名称 示例值：usertest // DTS任务名称
MigrateOption	是	MigrateOption	迁移任务配置选项
SrcDatabaseType	是	String	源实例数据库类型，目前支持：mysql，redis，mongodb，postgresql，mariadb，percona，sqlserver 不同地域数据库类型的具体支持情况，请参考控制台创建迁移页面。 示例值：mysql // 源实例是mysql
SrcAccessType	是	String	源实例接入类型，值包括：extranet(外网),cvm(CVM自建实例),dcg(专线接入的实例),vpncloud(云VPN接入的实例),cdb(腾讯云数据库实例),ccn(云联网实例) 示例值：extranet // 公网迁移
SrcInfo	是	SrcInfo	源实例信息，具体内容跟迁移任务类型相关
DstDatabaseType	是	String	目标实例数据库类型，目前支持：mysql，redis，mongodb，postgresql，mariadb，percona，sqlserver，cynosdbmysql。不同地域数据库类型的具体支持情况，请参考控制台创建迁移页面。 示例值：mysql // 目标实例mysql
DstAccessType	是	String	目标实例接入类型，目前支持：cdb（腾讯云数据库实例） 示例值：cdb // 目标实例是腾讯云mysql
DstInfo	是	DstInfo	目标实例信息
DatabaseInfo	否	String	需要迁移的源数据库表信息，用json格式的字符串描述。当MigrateOption.MigrateObject配置为2（指定库表迁移）时必填。 对于database-table两级结构的数据库： [{"Database":"db1","Table":["table1","table2"]},{"Database":"db2"}] 对于database-schema-table三级结构： [{"Database":"db1","Schema":"s1","Table":["table1","table2"]}, {"Database":"db1","Schema":"s2","Table":["table1","table2"]}, {"Database":"db2","Schema":"s1","Table":["table1","table2"]},{"Database":"db3"}, {"Database":"db4","Schema":"s1"}] 示例值：[{"Database":"db1","Table":["table1","table2"]},{"Database":"db2"}]
Tags.N	否	Array of TagItem	迁移实例的tag

参数名称	必选	类型	描述
SrcNodeType	否	String	源实例类型: ""或者"simple":主从节点，"cluster": 集群节点 示例值：cluster
SrcInfoMulti.N	否	Array of SrcInfo	源实例信息，具体内容跟迁移任务类型相关

3. 输出参数

参数名称	类型	描述
JobId	String	数据迁移任务ID 示例值：dts-1kl0iy0v
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建专线数据迁移任务

将用户通过专线接入的mysql实例, 全量+增量迁移迁移到腾讯云上海地域的cdb-d0dqi8nv实例上. 其中源实例的接入地域为广州.

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateMigrateJob
<公共请求参数>

{
"JobName": "设置为一个方便辨识的名称",
"SrcDatabaseType": "mysql",
"SrcAccessType": "dcg",
"SrcInfo": {
"Supplier": "others",
"UniqDcgId": "dcg-cyrjcc09",
"VpcId": "vpc-72jblfaa",
"SubnetId": "subnet-7raec42a",
"Ip": "192.168.120.136",
"Region": "ap-guangzhou",
"User": "root",
"Password": "yourPassword",
"Port": 3306
},
"DstDatabaseType": "mysql",
"DstAccessType": "cdb",
"DstInfo": {
"InstanceId": "cdb-d0dqi8nv",
"Region": "ap-shanghai",
"ReadOnly": 0,
"User": "root",
"Password": "yourPassword"
},
"DatabaseInfo": "",
"Tags": [
{
"TagKey": "负责人",
```

```
"TagValue": "bob"
}
],
"MigrateOption": {
"ExternParams": "",
"MigrateObject": 1,
"RunMode": 1,
"ExpectTime": "2020-09-22 00:00:00",
"ConsistencyType": 2,
"MigrateType": 2,
"IsOverrideRoot": 0
}
}
```

输出示例

```
{
"Response": {
"JobId": "dts-o3s1vxsp",
"RequestId": "915bc42a-714f-4faa-915b-a51cc09f5714"
}
}
```

示例2 创建云数据库迁移任务

从腾讯云mysql实例结构迁移到腾讯云另一个mysql实例，选择部分库表做，不做数据一致性检查。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateMigrateJob
<公共请求参数>

{
"JobName": "设置为一个方便辨识的名称",
"SrcDatabaseType": "mysql",
"SrcAccessType": "cdb",
"SrcInfo": {
"Supplier": "others",
"InstanceId": "cdb-ieow93923",
"User": "root",
"Password": "yourPassword"
},
"DstDatabaseType": "mysql",
"DstAccessType": "cdb",
"DstInfo": {
"InstanceId": "cdb-e78e0nnv",
"Region": "ap-shanghai",
"ReadOnly": 0,
"User": "root",
"Password": "yourPassword"
},
"DatabaseInfo": "",
"Tags": [
{
```

```
"TagKey": "负责人",
"TagValue": "bob"
}
],
"MigrateOption": {
"ExternParams": "",
"MigrateObject": 1,
"RunMode": 1,
"ExpectTime": "2020-09-22 00:00:00",
"ConsistencyType": 5,
"MigrateType": 2,
"IsOverrideRoot": 0
}
}
```

输出示例

```
{
"Response": {
"JobId": "dts-46i7easd",
"RequestId": "bc94c57b-9d69-11e9-84cb-256e968056b0"
}
}
```

示例3 创建公网数据迁移任务

将用户的mysql公网实例数据，采用全量方式迁移到腾讯云上海地域的cdb-e78e0nnv实例上。其中源实例的接入地域为广州。数据一致性检查做全量检测。

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateMigrateJob
<公共请求参数>

{
"JobName": "设置为一个方便辨识的名称",
"SrcDatabaseType": "mysql",
"SrcAccessType": "extranet",
"SrcInfo": {
"Supplier": "others",
"Ip": "14.17.22.36",
"Region": "ap-guangzhou",
"User": "root",
"Password": "yourPassword",
"Port": 3306
},
"DstDatabaseType": "mysql",
"DstAccessType": "cdb",
"DstInfo": {
"InstanceId": "cdb-e78e0nnv",
"Region": "ap-shanghai",
"ReadOnly": 0,
"User": "root",
"Password": "yourPassword"
},
```

```
"DatabaseInfo": "",
"Tags": [
{
"TagKey": "负责人",
"TagValue": "bob"
}
],
"MigrateOption": {
"ExternParams": "",
"MigrateObject": 1,
"RunMode": 1,
"ExpectTime": "2020-09-22 00:00:00",
"ConsistencyType": 2,
"MigrateType": 2,
"IsOverrideRoot": 0
}
}
```

输出示例

```
{
"Response": {
"JobId": "dts-1kl0iy0v",
"RequestId": "2201c42a-714f-4faa-915b-a51cc09f5cec"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.InternalHttpServerError	http请求访问出错。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.BizInvalidParameterValueError	参数值错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.BizInvalidParameterValueError	业务参数错误。
InvalidParameterValue.DataConvertError	数据转换错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
OperationDenied	操作被拒绝。
OperationDenied.NotEnoughMoneyError	用户余额不足。
OperationDenied.OperationDeniedError	操作被拒绝。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

创建校验迁移任务

最近更新时间：2025-04-25 01:29:06

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

创建校验迁移任务
在开始迁移前, 必须调用本接口创建校验, 且校验成功后才能开始迁移. 校验的结果可以通过DescribeMigrateCheckJob查看.
校验成功后,迁移任务若有修改, 则必须重新创建校验并通过后, 才能开始迁移.

如果是金融区链路, 请使用域名: https://dts.ap-shenzhen-fsi.tencentcloudapi.com

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateMigrateCheckJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建校验迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=CreateMigrateCheckJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "46b45da1-3e8e-4ef2-8de8-b1bffa027385"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.ProxyError	代理出错。
InternalError.AddTaskError	添加异步任务失败。
InternalError.CeleryError	内部调度系统错误。
InternalError.CgwSystemError	cgw系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.LockError	锁冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotFound	实例不存在。
InvalidParameter.InvalidIpAddress	ip地址错误。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
OperationDenied	操作被拒绝。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound.JobNotExist	迁移任务不存在。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

错误码	描述
UnsupportedOperation.IntraNetUserNotTaggedError	自研上云用户创建迁移任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。

完成数据迁移任务

最近更新时间：2025-04-25 01:29:06

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（CompleteMigrateJob）用于完成数据迁移任务。
选择采用增量迁移方式的任务，需要在迁移进度进入准备完成阶段后，调用本接口，停止迁移增量数据。
通过DescribeMigrateJobs接口查询到任务的状态为准备完成（status=8）时，此时可以调用本接口完成迁移任务。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CompleteMigrateJob。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
JobId	是	String	数据迁移任务ID 示例值：dts-1kl0iy0v
CompleteMode	否	String	完成任务的方式,仅支持旧版MySQL迁移任务。waitForSync-等待主从差距为0才停止,immediately-立即完成，不会等待主从差距一致。默认为waitForSync 示例值：waitForSync

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 完成数据迁移任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=CompleteMigrateJob
&JobId=dts-1kl0iy0v
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "5268ea15-6636-44d1-89f9-e989d1c82e7d"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError.CeleryError	内部调度系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
OperationDenied.JobOperationDeniedError	任务操作失败。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。

版权所有：腾讯云计算（北京）有限责任公司

数据订阅相关接口
修改订阅实例自动续费标识

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

修改订阅实例自动续费标识

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeAutoRenewFlag。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID，例如：subs-8uey736k 示例值：subs-8uey736k
AutoRenewFlag	是	Integer	自动续费标识。1-自动续费，0-不自动续费 示例值：1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置数据订阅自动续费标识

配置数据订阅自动续费标识

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifySubscribeAutoRenewFlag
&SubscribeId=subs-8392jd821u
&AutoRenewFlag=1
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
```

```
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InvalidParameter	参数错误。
InvalidParameter.InstanceNotFound	实例不存在。
ResourceNotFound	资源不存在。

重置数据订阅实例

最近更新时间：2025-04-25 01:29:02

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ResetSubscribe)用于重置数据订阅实例，已经激活的数据订阅实例，重置后可以使用ActivateSubscribe接口绑定其他的数据库实例

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ResetSubscribe。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-ieuwi83j2e

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 重置数据订阅实例

输入示例

```
https://dts.tencentcloudapi.com/?Action=ResetSubscribe
&SubscribeId=subs-ieuwi83j2e
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError	内部错误。
InvalidParameter	参数错误。
OperationDenied	操作被拒绝。
ResourceNotFound	资源不存在。

下线已隔离的数据订阅实例

最近更新时间：2025-04-25 01:29:02

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（OfflineIsolatedSubscribe）用于下线已隔离的数据订阅实例

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：OfflineIsolatedSubscribe。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-ieuwi83j2e

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 下线已隔离的数据订阅实例

订阅实例已经在隔离中，立即发起下线任务

输入示例

```
https://dts.tencentcloudapi.com/?Action=OfflineIsolatedSubscribe
&SubscribeId=subs-ieuwi83j2e
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.FailedOperationError	操作失败。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
FailedOperation.StatusInConflictError	当前状态不允许操作。
InternalError	内部错误。
InvalidParameter	参数错误。
ResourceNotFound	资源不存在。

修改数据订阅实例的IP和端口号

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeVipVport)用于修改数据订阅实例的IP和端口号

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeVipVport。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-jvxbbos7c0
DstUniqSubnetId	否	String	指定目的子网，如果传此参数，DstIp必须在目的子网内 示例值：subnet-njj6i45t
DstIp	否	String	目标IP，与DstPort至少传一个 示例值：10.0.0.37
DstPort	否	Integer	目标PORT，支持范围为：[1025-65535] 示例值：10086

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅实例的IP和端口号

输入示例

```
POST / HTTP/1.1
Host: dts.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ModifySubscribeName
<公共请求参数>

{
"SubscribeId": "subs-jvxbbos7c0",
"DstUniqSubnetId": "subnet-njj6i45t",
```

```
"DstIp": "10.0.0.37",
"DstPort": 10086
}
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError	内部错误。
InvalidParameter	参数错误。
ResourceNotFound	资源不存在。

修改数据订阅通道的订阅规则

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeObjects)用于修改数据订阅通道的订阅规则

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeObjects。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-ieuwi83j2e
SubscribeObjectType	是	Integer	数据订阅的类型，可选的值有：0 - 全实例订阅；1 - 数据订阅；2 - 结构订阅；3 - 数据订阅+结构订阅 示例值：0
Objects.N	否	Array of SubscribeObject	订阅的数据库表信息

3. 输出参数

参数名称	类型	描述
AsyncRequestId	String	异步任务的ID 示例值：19b514a7-816c43c1-ffb34ab6-8c6a23eb
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅通道的订阅规则

修改数据订阅的订阅规则，如添加或者移除某些库表

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifySubscribeObjects
&SubscribeId=subs-ieuwi83j2e
&SubscribeObjectType=0
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb",
"AsyncRequestId": "19b514a7-816c43c1-ffb34ab6-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFoundError	资源未找到。

修改数据订阅实例的名称

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeName)用于修改数据订阅实例的名称

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeName。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-ieuwi83j2e
SubscribeName	是	String	数据订阅实例的名称，长度限制为[1,60] 示例值：binlog订阅

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅实例的名称

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifySubscribeName
&SubscribeId=subs-ieuwi83j2e
&SubscribeName=binlog订阅
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
ResourceNotFound	资源不存在。

修改数据订阅实例通道的消费时间点

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(ModifySubscribeConsumeTime)用于修改数据订阅通道的消费时间点

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ModifySubscribeConsumeTime。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	数据订阅实例的ID 示例值：subs-jvxbbos7c0
ConsumeStartTime	是	String	消费时间起点，也即是指定订阅数据的时间起点，时间格式如：Y-m-d h:m:s，取值范围为过去24小时之内 示例值：2024-01-02 15:04:05

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 修改数据订阅通道的消费时间起点

重置数据订阅通道的数据起点，这样SDK可以消费到以此时间开始的数据

输入示例

```
https://dts.tencentcloudapi.com/?Action=ModifySubscribeConsumeTime
&SubscribeId=subs-ieuwi83j2e
&ConsumeStartTime=2019-10-26 00:00:00
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
FailedOperation.StatusInConflictError	当前状态不允许操作。
InternalError	内部错误。
InvalidParameter	参数错误。
OperationDenied	操作被拒绝。
ResourceNotFound	资源不存在。

隔离订阅实例

最近更新时间：2025-04-25 01:29:03

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（IsolateSubscribe）用于隔离小时计费的订阅实例。调用后，订阅实例将不能使用，同时停止计费。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：IsolateSubscribe。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID 示例值：subs-ieuwi83j2e

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 隔离订阅对象

输入示例

```
https://dts.tencentcloudapi.com/?Action=IsolateSubscribe
&SubscribeId=subs-ieuwi83j2e
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError.UnknownError	未知的内部错误。
InvalidParameter	参数错误。
InvalidParameter.BizInvalidParameterValueError	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
UnsupportedOperation	操作不支持。

版权所有：腾讯云计算（北京）有限责任公司

获取数据订阅实例列表

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(DescribeSubscribes)获取数据订阅实例信息列表，默认分页，每次返回20条

默认接口请求频率限制：100次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribes。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	否	String	数据订阅的实例ID 示例值：subs-fr19dk5e2y
SubscribeName	否	String	数据订阅的实例名称 示例值：name-fr19dk5e2y
InstanceId	否	String	绑定数据库实例的ID 示例值：cdb-5gsip99h
ChannelId	否	String	数据订阅实例的通道ID 示例值：topic-subs-fr19dk5e2y-cdb-5gsip99h
PayType	否	String	计费模式筛选，可能的值：0-包年包月，1-按量计费 示例值：1
Product	否	String	订阅的数据库产品，如mysql 示例值：mysql
Status.N	否	Array of String	数据订阅实例的状态，creating - 创建中，normal - 正常运行，isolating - 隔离中，isolated - 已隔离，offlining - 下线中 示例值：["isolated"]
SubsStatus.N	否	Array of String	数据订阅实例的配置状态，unconfigure - 未配置， configuring - 配置中，configured - 已配置 示例值：["running"]
Offset	否	Integer	返回记录的起始偏移量，默认为0。请输入非负整数 示例值：0
Limit	否	Integer	单次返回的记录数量，默认20。请输入1到100的整数 示例值：1
OrderDirection	否	String	排序方向，可选的值为"DESC"和"ASC"，默认为"DESC"，按创建时间逆序排序 示例值：DESC
TagFilters.N	否	Array of TagFilter	标签过滤条件
SubscribeVersion	否	String	订阅实例版本;txdts-旧版数据订阅，kafka-kafka版本数据订阅 示例值：kafka

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	符合查询条件的实例总数 示例值：2
Items	Array of SubscribeInfo	数据订阅实例的信息列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询数据订阅实例列表

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeSubscribes
&Status.0=isolated
&Offset=0
&Limit=1
&<公共请求参数>
```

输出示例


```
{
"Response": {
"TotalCount": 2,
"Items": [
{
"ChannelId": "dts-channel-C9rS58f2rFx11ZEZ",
"ConsumeStartTime": "2019-12-19 16:23:45",
"CreateTime": "2019-12-20 14:59:07",
"ExpireTime": "0000-00-00 00:00:00",
"InstanceId": "cdb-forsubsc",
"InstanceStatus": "running",
"IsolateTime": "0000-00-00 00:00:00",
"ModifyTime": "2019-12-20 18:06:26",
"OfflineTime": "0000-00-00 00:00:00",
"PayType": 1,
"Product": "mysql",
"Region": "ap-guangzhou",
"SdkConsumedTime": "0000-00-00 00:00:00",
"Status": "normal",
"SubsStatus": "configured",
"SubscribeId": "subs-58zcu8bax0",
"SubscribeName": "测试实例",
"UniqSubnetId": "subnet-4p1hpsj8",
"UniqVpcId": "vpc-fbg8gh1j",
"Vip": "192.168.1.143",
"Vport": 7507,
"SubscribeVersion": "kafka",
"AutoRenewFlag": 0,
"Tags": [
{
"TagKey": "负责人",
"TagValue": "jason"
```

```
}
]
}
],
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
InternalError	内部错误。
InternalError.InternalErrorError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。

查询订阅实例配置

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（DescribeSubscribeConf）用于查询订阅实例配置

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeSubscribeConf。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID 示例值：subs-ieuwi83j2e

3. 输出参数

参数名称	类型	描述
SubscribeId	String	订阅实例ID 示例值：subs-58zcu8bax0
SubscribeName	String	订阅实例名称 示例值：测试实例
ChannelId	String	订阅通道 示例值：dts-channel-C9rS58f2rFx11ZEZ
Product	String	订阅数据库类型 示例值：mysql
InstanceId	String	被订阅的实例 示例值：cdb-forsubsc
InstanceStatus	String	被订阅的实例的状态，可能的值有running,offline,isolate 示例值：running
SubsStatus	String	订阅实例状态，可能的值有unconfigure-未配置，configuring-配置中，configured-已配置 示例值：configured
Status	String	订阅实例生命周期状态，可能的值有：normal-正常，isolating-隔离中，isolated-已隔离，offlining-下线中 示例值：normal
CreateTime	String	订阅实例创建时间 示例值：2019-12-20 14:59:07
IsolateTime	String	订阅实例被隔离时间 示例值：0000-00-00 00:00:00

参数名称	类型	描述
ExpireTime	String	订阅实例到期时间 示例值：0000-00-00 00:00:00
OfflineTime	String	订阅实例下线时间 示例值：0000-00-00 00:00:00
ConsumeStartTime	String	订阅实例消费时间起点。 示例值：2019-12-19 16:23:45
PayType	Integer	订阅实例计费类型，1-小时计费，0-包年包月 示例值：1
Vip	String	订阅通道Vip 示例值：192.168.1.143
Vport	Integer	订阅通道Port 示例值：7507
UniqVpcId	String	订阅通道所在VpcId 示例值：vpc-fbg8gh1j
UniqSubnetId	String	订阅通道所在SubnetId 示例值：subnet-4p1hpsj8
SdkConsumedTime	String	当前SDK消费时间位点 示例值：0000-00-00 00:00:00
SdkHost	String	订阅SDK IP地址 示例值：192.168.1.143
SubscribeObjectType	Integer	订阅对象类型0-全实例订阅，1-DDL数据订阅，2-DML结构订阅，3-DDL数据订阅+DML结构订阅 示例值：3
SubscribeObjects	Array of SubscribeObject	订阅对象，当SubscribeObjectType 为0时，此字段为空数组
ModifyTime	String	修改时间 示例值：2019-12-20 18:06:26
Region	String	地域 示例值：ap-guangzhou
Tags	Array of TagItem	订阅实例的标签
AutoRenewFlag	Integer	自动续费标识,0-不自动续费，1-自动续费 示例值：0
SubscribeVersion	String	数据订阅版本。老版订阅填txdts，kafka版填kafka 示例值：txdts
Errors	Array of SubsErr	错误信息 注意：此字段可能返回 null，表示取不到有效值。
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查看订阅配置详情

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeSubscribeConf
&SubscribeId=subs-ieuwi83j2e
&<公共请求参数>
```

输出示例


```
{
"Response": {
"ChannelId": "dts-channel-C9rS58f2rFx11ZEZ",
"ConsumeStartTime": "2019-12-19 16:23:45",
"CreateTime": "2019-12-20 14:59:07",
"ExpireTime": "0000-00-00 00:00:00",
"InstanceId": "cdb-forsubsc",
"InstanceStatus": "running",
"IsolateTime": "0000-00-00 00:00:00",
"ModifyTime": "2019-12-20 18:06:26",
"OfflineTime": "0000-00-00 00:00:00",
"PayType": 1,
"Product": "mysql",
"Region": "ap-guangzhou",
"RequestId": "e580eac0-26c7-11ea-8e66-910ac0afe95e",
"SdkConsumedTime": "0000-00-00 00:00:00",
"SdkHost": "",
"Status": "normal",
"SubsStatus": "configured",
"SubscribeId": "subs-58zcu8bax0",
"SubscribeName": "测试实例",
"SubscribeObjectType": 3,
"SubscribeObjects": [
{
"DatabaseName": "dts",
"ObjectsType": 0,
"TableNames": []
}
],
"UniqSubnetId": "subnet-4p1hpsj8",
"SubscribeVersion": "kafka",
"UniqVpcId": "vpc-fbg8gh1j",
"Vip": "192.168.1.143",
"Vport": 7507,
"AutoRenewFlag": 0,
"Errors": [
{
"Message": ""
}
],
"Tags": [
{
"TagKey": "userDefineKey",
"TagValue": "userDefineValue"
}
]
}
}
```


5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue.InvalidParameterValue	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
ResourceNotFound	资源不存在。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。

查询任务执行结果

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口（DescribeAsyncRequestInfo）用于查询任务执行结果

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：DescribeAsyncRequestInfo。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
AsyncRequestId	是	String	任务 ID 示例值：cafebabe-254f-11ea-8995-e92c139e6918

3. 输出参数

参数名称	类型	描述
Info	String	任务执行结果信息 示例值：success
Status	String	任务执行状态，可能的值有：success，failed，running 示例值：ok
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 查询任务执行结果

查询任务执行结果

输入示例

```
https://dts.tencentcloudapi.com/?Action=DescribeAsyncRequestInfo
&AsyncRequestId=cafebabe-254f-11ea-8995-e92c139e6918
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Status": "success",
"Info": "ok",
```

```
"RequestId": "cafebabe-254f-11ea-8995-e92c139e6918"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError	内部错误。
InvalidParameter	参数错误。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。
UnsupportedOperation	操作不支持。

创建订阅对象

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口(CreateSubscribe)用于创建一个数据订阅实例。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer　点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：CreateSubscribe。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
Product	是	String	订阅的数据库类型，目前支持的有 mysql 示例值：mysql
PayType	是	Integer	实例付费类型，1小时计费，0包年包月 示例值：1
Duration	否	Integer	购买时长。PayType为0时必填。单位为月，最大支持120 示例值：120
Count	否	Integer	购买数量,默认为1，最大为10 示例值：1
AutoRenew	否	Integer	是否自动续费，0表示不自动续费，1表示自动续费，默认为0。小时计费实例设置该标识无效。 示例值：1
Tags.N	否	Array of TagItem	实例资源标签
Name	否	String	用户自定义实例名 示例值：user-defined-name

3. 输出参数

参数名称	类型	描述
SubscribeIds	Array of String	数据订阅实例的ID数组 示例值：["subs-ieuwi83j2e"]
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 创建数据订阅实例

输入示例

```
https://dts.tencentcloudapi.com/?Action=CreateSubscribe
&Product=mysql
&PayType=1
&<公共请求参数>
```

输出示例

```
{
"Response": {
"SubscribeIds": [
"subs-ieuwi83j2e"
],
"RequestId": "14a719b5-ffb34ab6-816c43c1-8c6a23eb"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError	内部错误。
InternalError.InternalTradeError	调用计费服务失败。
InvalidParameter	参数错误。
OperationDenied	操作被拒绝。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。

配置数据订阅通道

最近更新时间：2025-04-25 01:29:04

1. 接口描述

接口请求域名： dts.tencentcloudapi.com 。

本接口用于配置数据订阅，只有在未配置状态的订阅实例才能调用此接口。

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 公共请求参数。

参数名称	必选	类型	描述
Action	是	String	公共参数，本接口取值：ActivateSubscribe。
Version	是	String	公共参数，本接口取值：2018-03-30。
Region	是	String	公共参数，详见产品支持的 地域列表。
SubscribeId	是	String	订阅实例ID。 示例值：subs-ieuwi83j2e
InstanceId	是	String	数据库实例ID 示例值：cdb-9dijwkkw2
SubscribeObjectType	是	Integer	数据订阅类型0-全实例订阅，1数据订阅，2结构订阅，3数据订阅与结构订阅 示例值：0
Objects	否	SubscribeObject	订阅对象
UniqSubnetId	否	String	数据订阅服务所在子网。默认为数据库实例所在的子网内。 示例值：Default-VPC
Vport	否	Integer	订阅服务端口；默认为7507 示例值：7507

3. 输出参数

参数名称	类型	描述
AsyncRequestId	String	配置数据订阅任务ID。 示例值：cafebabe-254f-11ea-8995-e92c139e6978
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 配置数据订阅

配置数据订阅

输入示例

```
https://dts.tencentcloudapi.com/?Action=ActivateSubscribe
&SubscribeId=subs-ieuwi83j2e
```

```
&InstanceId=cdb-9dijwkkw2
&SubscribeObjectType=0
&<公共请求参数>
```

输出示例

```
{
"Response": {
"AsyncRequestId": "cafebabe-254f-11ea-8995-e92c139e6978",
"RequestId": "9iu7ueht-c38f-11ea-84e5-d5a93c1b919b"
}
}
```

5. 开发者资源

腾讯云 API 平台

腾讯云 API 平台 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

API Inspector

用户可通过 API Inspector 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 API Explorer 进行在线调试。

SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Java: GitHub, Gitee
- Tencent Cloud SDK 3.0 for PHP: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Go: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Node.js: GitHub, Gitee
- Tencent Cloud SDK 3.0 for .NET: GitHub, Gitee
- Tencent Cloud SDK 3.0 for C++: GitHub, Gitee
- Tencent Cloud SDK 3.0 for Ruby: GitHub, Gitee

命令行工具

- Tencent Cloud CLI 3.0

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 公共错误码。

错误码	描述
InternalError	内部错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InvalidParameter	参数错误。
InvalidParameter.InstanceNotFound	实例不存在。
InvalidParameterValue.DataConvertError	数据转换错误。
OperationDenied	操作被拒绝。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
ResourceNotFound	资源不存在。

错误码	描述
UnsupportedOperation	操作不支持。

数据结构

最近更新时间：2025-05-22 01:17:03

ConsistencyParams

抽样检验时的抽样参数

被如下接口引用：CreateMigrateJob, DescribeMigrateJobs, ModifyMigrateJob。

名称	类型	必选	描述
SelectRowsPerTable	Integer	是	数据内容检测参数。表中选出用来数据对比的行，占表的总行数的百分比。取值范围是整数[1-100] 示例值：10
TablesSelectAll	Integer	是	数据内容检测参数。迁移库表中，要进行数据内容检测的表，占所有表的百分比。取值范围是整数[1-100] 示例值：10
TablesSelectCount	Integer	是	数据数量检测，检测表行数是否一致。迁移库表中，要进行数据数量检测的表，占所有表的百分比。取值范围是整数[1-100] 示例值：10

DstInfo

目的实例信息，具体内容跟迁移任务类型相关

被如下接口引用：CreateMigrateJob, DescribeMigrateJobs, ModifyMigrateJob。

名称	类型	必选	描述
Region	String	是	目标实例地域，如ap-guangzhou 示例值：ap-guangzhou
InstanceId	String	否	目标实例ID，如cdb-jd92ijd8 示例值：cdb-rx7stlmc
ReadOnly	Integer	否	目前只对MySQL有效。当为整实例迁移时，1-只读，0-可读写。 示例值：1
User	String	否	目标数据库账号 示例值：dts
Password	String	否	目标数据库密码 示例值：dts123456789

ErrorInfo

迁移任务错误信息及提示

被如下接口引用：DescribeMigrateJobs。

名称	类型	描述
ErrorLog	String	具体的报错日志, 包含错误码和错误信息 示例值：invalid connection
HelpDoc	String	报错对应的帮助文档Ur 示例值：http://api.qcloud.com

MigrateDetailInfo

描述详细迁移过程

被如下接口引用：DescribeMigrateJobs。

名称	类型	描述
StepAll	Integer	总步骤数 示例值：3
StepNow	Integer	当前步骤 示例值：3
Progress	String	总进度,如："10" 示例值：30

名称	类型	描述
CurrentStepProgress	String	当前步骤进度,如:"1" 示例值：10
MasterSlaveDistance	Integer	主从差距，MB；在增量同步阶段有效，目前支持产品为：redis和mysql 示例值：10
SecondsBehindMaster	Integer	主从差距，秒；在增量同步阶段有效，目前支持产品为：mysql 示例值：100
StepInfo	Array of MigrateStepDetailInfo	步骤信息

MigrateJobInfo

迁移任务详情

被如下接口引用：DescribeMigrateJobs。

名称	类型	描述
JobId	String	数据迁移任务ID 示例值：dts-75sekyal
JobName	String	数据迁移任务名称 示例值：dts-75sekyal
MigrateOption	MigrateOption	迁移任务配置选项 示例值：{"ConsistencyParams":{"SelectRowsPerTable":0,"TablesSelectAll":0
SrcDatabaseType	String	源实例数据库类型:mysql，redis，mongodb，postgresql，mariadb，percona 示例值：mysql
SrcAccessType	String	源实例接入类型，值包括：extranet(外网),cvm(cvm自建实例),dcg(专线接入的实例),vpncloud(云vpn接入的实例),cdb(腾讯云数据库实例),ccn(云联网实例) 示例值：cdb
SrcInfo	SrcInfo	源实例信息，具体内容跟迁移任务类型相关 示例值：{"AccessKey":"","CcnId":"","CvmInstanceId":"","EngineVersion":""
DstDatabaseType	String	目标实例数据库类型:mysql，redis，mongodb，postgresql，mariadb，percona 示例值：mysql
DstAccessType	String	目标实例接入类型，目前支持：cdb(腾讯云数据库实例) 示例值：cdb
DstInfo	DstInfo	目标实例信息 示例值：{"CcnId":"","CvmInstanceId":"","InstanceId":"crs-eciqrvyo","Ip":
DatabaseInfo	String	需要迁移的源数据库表信息，如果需要迁移的是整个实例，该字段为[] 示例值：[{"Database":"db1","Table":["table1","table2"]},{"Database":"db2"}]
CreateTime	Timestamp	任务创建(提交)时间 示例值：2022-04-14 16:35:23
StartTime	Timestamp	任务开始执行时间 示例值：2022-04-14 16:48:33
EndTime	Timestamp	任务执行结束时间 示例值：2022-04-14 16:48:33
Status	Integer	任务状态,取值为：1-创建中(Creating),3-校验中(Checking)4-校验通过(CheckPass),5-校验不通过（CheckNotPass）,7-任务运行(Running),8-准备完成（ReadyComplete）,9-任务成功（Success）,10-任务失败（Failed）,11-撤销中（Stopping）,12-完成中（Completing）,23-未知状态（Unknown） 示例值：6
Detail	MigrateDetailInfo	任务详情 示例值：{"CurrentStepProgress":"0","MasterSlaveDistance":-1,"Progress":"

名称	类型	描述
ErrorInfo	Array of ErrorInfo	任务错误信息提示，当任务发生错误时，不为null或者空值
Tags	Array of TagItem	标签
SrcInfoMulti	Array of SrcInfo	源实例为集群时且接入为非cdb时源实例信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"AccessKey":"","CcnId":"","CvmInstanceId":"","EngineVersion":"

MigrateOption

迁移任务配置选项

被如下接口引用：CreateMigrateJob, DescribeMigrateJobs, ModifyMigrateJob。

名称	类型	必选	描述
RunMode	Integer	是	任务运行模式，值包括：1-立即执行，2-定时执行 示例值：1
ExpectTime	Timestamp	否	期望执行时间，当runMode=2时，该字段必填，时间格式：yyyy-mm-dd hh:mm:ss 示例值：2024-12-20 19:02:01
MigrateType	Integer	否	数据迁移类型，值包括：1-结构迁移,2-全量迁移,3-全量+增量迁移 示例值：3
MigrateObject	Integer	否	迁移对象，1-整个实例，2-指定库表 示例值：1
ConsistencyType	Integer	否	抽样数据一致性检测参数，1-未配置,2-全量检测,3-抽样检测, 4-仅校验不一致表,5-不检测 示例值：2
IsOverrideRoot	Integer	否	是否用源库Root账户覆盖目标库，值包括：0-不覆盖，1-覆盖，选择库表或者结构迁移时应该为0 示例值：0

ExternParams	String	否	不同数据库用到的额外参数.以JSON格式描述.
			Redis可定义如下的参数:
			{
			"ClientOutputBufferHardLimit":512, 从机缓冲区的硬性容量限制(MB)
			"ClientOutputBufferSoftLimit":512, 从机缓冲区的软性容量限制(MB)
			"ClientOutputBufferPersistTime":60, 从机缓冲区的软性限制持续时间(秒)
			"ReplBacklogSize":512, 环形缓冲区容量限制(MB)
			"ReplTimeout":120， 复制超时时间(秒)
			}
			MongoDB可定义如下的参数:
			{
			'SrcAuthDatabase':'admin',
			'SrcAuthFlag': "1",
			'SrcAuthMechanism':"SCRAM-SHA-1"
			}
			MySQL暂不支持额外参数设置。
			示例值：{"ClientOutputBufferHardLimit":512}

ConsistencyParams	ConsistencyParams	否	仅用于“抽样数据一致性检测”，ConsistencyType配置为抽样检测时，必选

MigrateStepDetailInfo

迁移中的步骤信息

被如下接口引用：DescribeMigrateJobs。

名称	类型	描述
StepNo	Integer	步骤序列 示例值：1

名称	类型	描述
StepName	String	步骤展现名称 示例值：全量导出
StepId	String	步骤英文标识 示例值：dumper
Status	Integer	步骤状态:0-默认值,1-成功,2-失败,3-执行中,4-未执行 示例值：3
StartTime	String	当前步骤开始的时间，格式为"yyyy-mm-dd hh:mm:ss"，该字段不存在或者为空是无意义 示例值：2024-12-20 19:02:01

SrcInfo

源实例信息

被如下接口引用：CreateMigrateJob, DescribeMigrateJobs, ModifyMigrateJob。

名称	类型	必选	描述
AccessKey	String	否	阿里云AccessKey。源库是阿里云RDS5.6适用 示例值：aBcD**Z012345
Ip	String	否	实例的IP地址 示例值：192.168.0.12
Port	Integer	否	实例的端口 示例值：3306
User	String	否	实例的用户名 示例值：dts
Password	String	否	实例的密码 示例值：password123
RdsInstanceId	String	否	阿里云RDS实例ID。源库是阿里云RDS5.6/5.6适用 示例值：rds-dgyfhvf
CvmInstanceId	String	否	CVM实例短ID，格式如：ins-olgl39y8，与云服务器控制台页面显示的实例ID相同。如果是CVM自建实例，需要传递此字段 示例值：ins-olgl39y8
UniqDcgId	String	否	专线网关ID，格式如：dcg-0rxtqqxb 示例值：dcg-0rxtqqxb
VpcId	String	否	私有网络ID，格式如：vpc-92jblxto 示例值：vpc-92jblxto
SubnetId	String	否	私有网络下的子网ID，格式如：subnet-3paxmkdz 示例值：subnet-3paxmkdz
UniqVpnGwId	String	否	VPN网关ID，格式如：vpngw-9ghexg7q 示例值：vpngw-9ghexg7q
InstanceId	String	否	数据库实例ID，格式如：cdb-powiqx8q 示例值：cdb-powiqx8q
Region	String	否	地域英文名，如：ap-guangzhou 示例值：ap-guangzhou
Supplier	String	否	当实例为RDS实例时，填写为aliyun, 其他情况均填写others 示例值：others
CcnId	String	否	云联网ID，如：ccn-afp6kltc 示例值：ccn-afp6kltc
EngineVersion	String	否	数据库版本，当实例为RDS实例时才有效，格式如：5.6或者5.7，默认为5.6 示例值：5.7

SubsErr

查询订阅配置的错误信息

被如下接口引用：DescribeSubscribeConf。

名称	类型	必选	描述
Message	String	否	错误信息 示例值：error

SubscribeInfo

订阅实例信息

被如下接口引用：DescribeSubscribes。

名称	类型	必选	描述
SubscribeId	String	否	数据订阅的实例ID 示例值：subs-9jyki7hniw
SubscribeName	String	否	数据订阅实例的名称 示例值：binlog订阅
ChannelId	String	否	数据订阅实例绑定的通道ID。kafka版订阅就是kafka topic 示例值：topic-subs-l4d3a7izai-cdb-24ifcx43
Product	String	否	订阅实例的类型，目前支持cynosdbmysql,mariadb,mongodb,mysql,percona,tdpg,tdsqlpercona(tdsqlmysql) 示例值：mysql
InstanceId	String	否	数据订阅实例绑定的数据库实例ID 示例值：cdb-24ifcx43
InstanceStatus	String	否	云数据库状态：running 运行中，isolated 已隔离，offline 已下线。如果不是云上，此值为空 示例值：running
SubsStatus	String	否	数据订阅状态，可能的值为：未启动 notStarted, 校验中 checking, 校验不通过 checkNotPass, 校验通过 checkPass, 启动中 starting, 运行中 running, 异常出错 error 示例值：running
ModifyTime	String	否	上次修改时间，时间格式如：Y-m-d h:m:s 示例值：2023-05-15 21:29:05
CreateTime	String	否	创建时间 示例值：2023-05-15 21:29:05
IsolateTime	String	否	隔离时间，时间格式如：Y-m-d h:m:s 示例值：2023-05-15 21:29:05
ExpireTime	String	否	包年包月任务的到期时间，时间格式如：Y-m-d h:m:s。默认：0000-00-00 00:00:00 示例值：2023-05-15 21:29:05
OfflineTime	String	否	下线时间 示例值：2023-05-15 21:29:05
ConsumeStartTime	String	否	最近一次修改的消费时间起点，如果从未修改则为零值 示例值：2023-05-15 21:29:05
AutoRenewFlag	Integer	否	自动续费标识。只有当 PayType=0，该值才有意义。枚举值：0-不自动续费，1-自动续费 示例值：0
Region	String	否	数据订阅实例所属地域 示例值：ap-guangzhou
PayType	Integer	否	计费方式，0 – 包年包月，1 – 按量计费 示例值：0
Vip	String	否	旧版订阅通道的vip 示例值：10.0.0.37

名称	类型	必选	描述
Vport	Integer	否	数据订阅实例的Vport 示例值：10086
UniqVpcId	String	否	数据订阅实例Vip所在VPC的唯一ID 示例值：vpc-3l4t891i
UniqSubnetId	String	否	数据订阅实例Vip所在子网的唯一ID 示例值：subnet-njj6i45t
Status	String	否	数据订阅生命周期状态，可能的值为：正常 normal, 隔离中 isolating, 已隔离 isolated, 下线中 offlining, 按量转包年包月中 post2PrePayIng 示例值：normal
SdkConsumedTime	String	否	SDK最后一条确认消息的时间戳，如果SDK一直消费，也可以作为SDK当前消费时间点 示例值：2023-05-15 21:29:05
Tags	Array of TagItem	否	标签 注意：此字段可能返回 null，表示取不到有效值。
SubscribeVersion	String	否	订阅实例版本；txdts-旧版数据订阅,kafka-kafka版本数据订阅 示例值：kafka

SubscribeObject

数据订阅的对象

被如下接口引用：ActivateSubscribe, DescribeSubscribeConf, ModifySubscribeObjects。

名称	类型	必选	描述
ObjectsType	Integer	是	数据订阅对象的类型，0-数据库，1-数据库内的表 示例值：1
DatabaseName	String	是	订阅数据库的名称 示例值：db_1
TableNames	Array of String	否	订阅数据库中表名称数组 示例值：["tab_1"]

TagFilter

标签过滤

被如下接口引用：DescribeMigrateJobs, DescribeSubscribes。

名称	类型	必选	描述
TagKey	String	是	标签键值 示例值：key
TagValue	Array of String	否	标签值 示例值：[value1, value2]

TagItem

标签

被如下接口引用：CreateMigrateJob, CreateSubscribe, DescribeMigrateJobs, DescribeSubscribeConf, DescribeSubscribes。

名称	类型	必选	描述
TagKey	String	是	标签键值 示例值：key
TagValue	String	否	标签值 示例值：value

错误码

最近更新时间：2024-12-20 01:25:46

功能说明

如果返回结果中存在 Error 字段，则表示调用 API 接口失败。例如：

```
{
"Response": {
"Error": {
"Code": "AuthFailure.SignatureFailure",
"Message": "The provided credentials could not be validated. Please check your signature is correct."
},
"RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
}
}
```

Error 中的 Code 表示错误码，Message 表示该错误的具体信息。

错误码列表

公共错误码

错误码	说明
ActionOffline	接口已下线。
AuthFailure.InvalidAuthorization	请求头部的 `Authorization` 不符合腾讯云标准。
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。
AuthFailure.MFAFailure	MFA 错误。
AuthFailure.SecretIdNotFound	密钥不存在。请在 控制台 检查密钥是否已被删除或者禁用，如状态正常，请检查密钥是否填写正确，注意前后不得有空格。
AuthFailure.SignatureExpire	签名过期。Timestamp 和服务器时间相差不得超过五分钟，请检查本地时间是否和标准时间同步。
AuthFailure.SignatureFailure	签名错误。签名计算错误，请对照调用方式中的签名方法文档检查签名计算过程。
AuthFailure.TokenFailure	token 错误。
AuthFailure.UnauthorizedOperation	请求未授权。请参考 CAM 文档对鉴权的说明。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidAction	接口不存在。
InvalidParameter	参数错误（包括参数格式、类型等错误）。
InvalidParameterValue	参数取值错误。
InvalidRequest	请求 body 的 multipart 格式错误。
IpInBlacklist	IP 地址在黑名单中。
IpNotInWhitelist	IP 地址不在白名单中。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数。
NoSuchProduct	产品不存在
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
RequestLimitExceeded.GlobalRegionUinLimitExceeded	主账号超过频率限制。

错误码	说明
RequestLimitExceeded.IPLimitExceeded	IP 限频。
RequestLimitExceeded.UinLimitExceeded	主账号限频。
RequestSizeLimitExceeded	请求包超过限制大小。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResponseSizeLimitExceeded	返回包超过限制大小。
ServiceUnavailable	当前服务暂时不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误，用户多传未定义的参数会导致错误。
UnsupportedOperation	操作不支持。
UnsupportedProtocol	http(s) 请求协议错误，只支持 GET 和 POST 请求。
UnsupportedRegion	接口不支持所传地域。

业务错误码

错误码	说明
AuthFailure.AuthFailureError	认证失败。
AuthFailure.UnauthorizedOperationError	鉴权失败，当前用户不允许执行该操作。
FailedOperation.FailedOperationError	操作失败。
FailedOperation.NotAllowOperation	禁止该操作。
FailedOperation.ProxyError	代理出错。
FailedOperation.StartJobFailed	任务开始失败。
FailedOperation.StatusInConflict	当前状态冲突，不能执行该操作。
FailedOperation.StatusInConflictError	当前状态不允许操作。
InternalError.AddTaskError	添加异步任务失败。
InternalError.CeleryError	内部调度系统错误。
InternalError.CgwSystemError	cgw系统错误。
InternalError.DatabaseError	迁移平台数据库访问失败。
InternalError.DuplicateJob	迁移任务冲突。
InternalError.InternalErrorError	内部错误。
InternalError.InternalHttpServerError	http请求访问出错。
InternalError.InternalInnerCommonError	内部组件访问错误。
InternalError.InternalTradeError	调用计费服务失败。
InternalError.LockError	锁冲突。
InternalError.ProtocolError	通信协议错误。
InternalError.UndefinedError	内部错误。

错误码	说明
InternalError.UnknownError	未知的内部错误。
InvalidParameter.BizInvalidParameterValueError	参数值错误。
InvalidParameter.InstanceNotFound	实例不存在。
InvalidParameter.InvalidIpAddress	ip地址错误。
InvalidParameter.InvalidParameterError	参数无效。
InvalidParameterValue.BizInvalidParameterValueError	业务参数错误。
InvalidParameterValue.DataConvertError	数据转换错误。
InvalidParameterValue.InvalidParameterValue	参数值错误。
InvalidParameterValue.InvalidParameterValueError	非法参数。
LimitExceeded.MaxUnusedJobs	闲置迁移任务数目超过限制。
OperationDenied	操作被拒绝。
OperationDenied.BizOperationDeniedError	当前操作不满足条件被阻止。
OperationDenied.JobOperationDeniedError	任务操作失败。
OperationDenied.MigrateServiceSupportError	数据迁移服务不支持当前迁移类型。
OperationDenied.NotEnoughMoneyError	用户余额不足。
OperationDenied.OperationDenied	该操作不能执行。
OperationDenied.OperationDeniedError	操作被拒绝。
RequestLimitExceeded.RequestLimitExceededError	请求次数超限。
ResourceNotFound.BizResourceNotFoundError	未找到资源。
ResourceNotFound.JobNotExist	迁移任务不存在。
ResourceNotFound.ResourceNotFound	实例未找到。
ResourceNotFound.ResourceNotFoundError	资源未找到。
UnauthorizedOperation.NotEnoughPrivileges	认证失败，没有足够权限。
UnauthorizedOperation.UnauthorizedOperationError	未授权的操作。
UnsupportedOperation.IntraNetUserNotTaggedError	自研上云用户创建迁移任务时未指定标签，需要补齐“运营部门”、“运营产品”、“负责人”这三类标签。

版权所有：腾讯云计算（北京）有限责任公司