Elasticsearch Service

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
- 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
- 实例相关接口
  - 创建索引
  - 创建ES集群实例
  - 删除索引
  - 销毁ES集群实例
  - 查询智能运维诊断结果报告
  - 查询ES集群日志
  - 查询实例插件列表
  - 智能运维诊断集群
  - 查看智能运维配置
  - 获取接收客户端请求的节点类型
  - 集群续费询价
  - 修改集群VIP绑定的安全组
  - 重启ES集群实例
  - 重启Kibana
  - 重启集群节点
  - 更新智能运维配置
  - 更新ES集群词典
  - 更新索引
  - 更新ES集群实例
  - 更新实例Jdk配置
  - 变更插件列表
  - 更新接收客户端请求的节点类型
  - 升级ES集群版本
  - 升级ES商业特性
  - 获取索引列表
  - 获取索引元数据
  - 查询实例操作记录
  - 查询ES集群实例
  - 查询集群视图
  - 安装ES集群模型
  - 快照备份恢复
  - 获取快照备份列表
  - 删除快照仓库里备份的快照
  - 创建集群快照
- Serverless实例相关接口
  - 创建Serverless索引项目空间(new)
  - 删除Serverless索引
  - 更新Serverless索引空间
  - 创建Serverless索引
  - 查看Serverless空间子用户
  - 删除Serverless空间子用户
  - Serverless获取索引列表
  - 获取Serverless索引空间列表

- 更新Serverless索引
- 获取实例对应的监控指标
- space维度的kibana获取登录token
- 查询快照信息接口
- cos迁移流程
- 检查cos迁移索引元数据

Logstash相关接口
- 创建Logstash实例
- 删除Logstash实例
- 删除Logstash管道
- 查询Logstash实例日志
- 获取Logstash实例列表
- 重启Logstash实例
- 保存并部署管道
- 启动Logstash管道
- 停止Logstash管道
- 更新Logstash实例
- 查询Logstash实例操作记录
- 获取Logstash实例管道列表
- 更新管道描述

数据结构

错误码

Elasticsearch Service API 2025-01-01
- 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
- 智能搜索开发相关接口
  - 实时文档解析
  - 文档解析（异步）
  - 获取文档解析结果
  - 在线文本切片
  - 文档切片（异步）
  - 获取文档切片结果
  - 获取特征向量
  - 重排序
  - 大模型生成
- 数据结构
- 错误码

API 文档
更新历史

最近更新时间：2025-06-24 01:24:15

第 83 次发布

发布时间：2025-06-24 01:24:06

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- CosBackup
  - 新增成员：EsRepositoryType, UserEsRepository, StorageDuration, AutoBackupInterval
- Snapshots
  - 新增成员：Repository

第 82 次发布

发布时间：2025-05-23 01:18:18

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- LogstashNodeInfo
  - 新增成员：Zone

第 81 次发布

发布时间：2025-05-21 01:18:51

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateLogstashInstance
  - 新增入参：MultiZoneInfo, DeployMode

修改数据结构：

- LogstashInstanceInfo
  - 新增成员：DeployMode, MultiZoneInfo
- ZoneDetail
  - 新增成员：Hidden

第 80 次发布

发布时间：2025-04-22 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：

- RestoreClusterSnapshot
  - **修改入参：** Password

第 79 次发布

发布时间：2025-04-11 01:32:41

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- DiDataSourceCvmInstance
  - **修改成员：** InstanceId, VpcId, SubnetId, ErrMsg
- InstanceLog
  - **修改成员：** NodeID
- KibanaView
  - **修改成员：** NodeId
- LogstashPipeline
  - **修改成员：** PipelineId, PipelineDesc, Config, Workers, BatchSize, BatchDelay, QueueType, QueueMaxBytes, QueueCheckPointWrites
- LogstashPipelineInfo
  - 新增成员：CreateTime, UpdateTime
- Snapshots
  - 新增成员：UserBackUp

第 78 次发布

发布时间：2025-03-14 01:16:35

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessIndexMetaField
  - 新增成员：IndexMetaJson

第 77 次发布

发布时间：2025-03-07 01:16:35

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessIndexSettingsField
  - 新增成员：CustomSetting

第 76 次发布

发布时间：2025-02-20 01:36:22

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- TaskDetail
  - **修改成员：** ProcessInfo

第 75 次发布

发布时间：2025-01-13 01:16:10

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- IndexMetaField
  - 新增成员：IndexAliasesField

第 74 次发布

发布时间：2024-12-11 01:33:45

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：EnableDestroyProtection

修改数据结构：

- EsDictionaryInfo
  - 新增成员：AnsjMain, AnsjStop, AnsjAmbiguity, AnsjSynonyms
- InstanceInfo
  - 新增成员：EnableScheduleRecoverGroup, EnableScheduleOperationDuration, EnableDestroyProtection
- Operation
  - 新增成员：RollbackTag
  - **修改成员：** SubAccountUin

第 73 次发布

发布时间：2024-11-15 01:34:50

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- InstallInstanceModel
  - 新增入参：ModelNames, TaskTypes

第 72 次发布

发布时间：2024-11-05 01:40:59

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessIndexOptionsField
  - 新增成员：StandardStorageAge

第 71 次发布

发布时间：2024-10-30 01:33:25

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateClusterSnapshot
- DeleteClusterSnapshot
- DescribeClusterSnapshot
- RestoreClusterSnapshot

新增数据结构：

- Failures
- Snapshots

第 70 次发布

发布时间：2024-10-17 01:18:44

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：EnableScheduleRecoverGroup, EnableScheduleOperationDuration
- RestartNodes
  - 新增入参：EnableScheduleRecoverGroup, EnableScheduleOperationDuration
- UpdateInstance
  - 新增入参：EnableScheduleRecoverGroup, EnableScheduleOperationDuration
- UpgradeInstance
  - 新增入参：EnableScheduleRecoverGroup, EnableScheduleOperationDuration

新增数据结构：

- EnableScheduleOperationDuration

修改数据结构：

- NodeView
  - 新增成员：CVMDisasterRecoverGroupId, CVMDisasterRecoverGroupStatus

第 69 次发布

发布时间：2024-09-10 15:12:32

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeSpaceKibanaTools

第 68 次发布

发布时间：2024-09-10 01:35:22

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeServerlessMetrics
  - 新增入参：BatchIndexList
  - 新增出参：MetricMapList

新增数据结构：

- MetricAllData
- MetricMapByIndexId

第 67 次发布

发布时间：2024-09-03 01:47:13

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- InstallInstanceModel

第 66 次发布

发布时间：2024-08-27 01:44:02

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：ReadWriteMode
- UpdateInstance
  - 新增入参：ReadWriteMode

修改数据结构：

- InstanceInfo
  - 新增成员：CosBucketStorageSize, ReadWriteMode

第 65 次发布

发布时间：2024-08-14 01:44:38

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateServerlessSpaceV2
  - 新增入参：TagList
- DescribeServerlessMetrics
  - 新增入参：DurationType
  - 新增出参：IndexTraffic, ReadReqTimes, WriteReqTimes, DocCount
- DescribeServerlessSpaces
  - 新增入参：TagList

修改数据结构：

- ServerlessIndexMetaField
  - 新增成员：IndexTraffic
- ServerlessSpace
  - 新增成员：TagList

第 64 次发布

发布时间：2024-08-08 01:41:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：SubProductCode
- DescribeInstances
  - 新增入参：CdcId

修改数据结构：

- InstanceInfo
  - 新增成员：SubProductCode

第 63 次发布

发布时间：2024-07-22 01:39:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- RestartNodes
  - 新增入参：ShardAllocationConcurrents, ShardAllocationBytes
- UpgradeInstance
  - 新增入参：ShardAllocationConcurrents, ShardAllocationBytes

第 62 次发布

发布时间：2024-07-17 01:15:53

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：ShardAllocationConcurrents, ShardAllocationBytes

第 61 次发布

发布时间：2024-07-09 01:14:44

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessIndexOptionsField
  - 新增成员：SinkCycleAge

第 60 次发布

发布时间：2024-07-05 18:47:21

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeServerlessMetrics

修改接口：

- CreateInstance
  - 新增入参：DisasterRecoverGroupAffinity

修改数据结构：

- InstanceInfo
  - 新增成员：DisasterRecoverGroupAffinity

第 59 次发布

发布时间：2024-06-14 01:50:54

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessSpace
  - 新增成员：ClusterType

第 58 次发布

发布时间：2024-05-14 01:51:30

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ServerlessSpace
  - 新增成员：KibanaLanguage

第 57 次发布

发布时间：2024-04-19 01:17:21

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeServerlessInstances

新增数据结构：

- ServerlessIndexMetaField
- ServerlessIndexNetworkField
- ServerlessIndexOptionsField
- ServerlessIndexSettingsField

第 56 次发布

发布时间：2024-04-11 16:23:06

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateServerlessInstance
  - **修改入参：** Zone, VpcId, SubnetId

第 55 次发布

发布时间：2024-03-28 10:47:55

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateIndex
  - **修改入参：** IndexMetaJson
- RestartNodes
  - 新增入参：CvmDelayOnlineTime
- UpdateInstance
  - 新增入参：OutboundPublicAcls, OutboundPublicAccess, CvmDelayOnlineTime

- UpgradeInstance
  - 新增入参：CvmDelayOnlineTime

新增数据结构：

- OutboundPublicAcl

修改数据结构：

- InstanceInfo
  - 新增成员：OutboundPublicAcls, NetConnectScheme
- NodeView
  - 新增成员：CVMStatus
  - **修改成员：** IsCoordinationNode

第 54 次发布

发布时间：2024-03-14 01:15:53

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CheckMigrateIndexMetaData
- CreateCosMigrateToServerlessInstance
- DescribeUserCosSnapshotList

新增数据结构：

- CommonIndexInfo
- CosSnapShotInfo
- DataStreamInfo

第 53 次发布

发布时间：2024-03-13 16:49:04

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateServerlessInstance
- CreateServerlessSpaceV2
- DeleteServerlessInstance
- DeleteServerlessSpaceUser
- DescribeServerlessSpaceUser
- DescribeServerlessSpaces
- UpdateServerlessInstance
- UpdateServerlessSpace

新增数据结构：

- DiData
- DiDataSinkServerless
- DiDataSourceCvm
- DiDataSourceCvmInstance
- DiDataSourceTke
- DiSourceCvm
- DiSourceTke
- DiSourceTkePodLabel
- KibanaPublicAcl

- ServerlessDi
- ServerlessSpace
- ServerlessSpaceUser
- VpcInfo

第 52 次发布

发布时间：2024-01-25 01:13:20

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：Protocol

第 51 次发布

发布时间：2023-12-05 01:46:13

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- InquirePriceRenewInstance

第 50 次发布

发布时间：2023-11-23 01:18:36

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- RestartInstance
  - 新增入参：UpgradeKernel
- UpdateInstance
  - 新增入参：KibanaPrivateDomain, CerebroPrivateDomain

修改数据结构：

- InstanceInfo
  - 新增成员：KibanaPrivateVip, CustomKibanaPrivateUrl
- OptionalWebServiceInfo
  - 新增成员：PrivateVip, CustomPrivateUrl

第 49 次发布

发布时间：2023-10-30 00:42:05

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeDiagnose
- DescribeInstancePluginList
- GetDiagnoseSettings

新增数据结构：

- DescribeInstancePluginInfo
- DiagnoseJobMeta
- DiagnoseJobResult
- DiagnoseResult
- Dimension
- JobParam
- LogDetail
- Metric
- MetricDetail
- SettingDetail

第 48 次发布

发布时间：2023-10-12 01:57:01

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- NodeView
  - 新增成员：IsCoordinationNode

第 47 次发布

发布时间：2023-10-09 01:15:32

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- IndexMetaField
  - 新增成员：IndexMetaJson

第 46 次发布

发布时间：2023-09-14 01:45:31

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- ProcessDetail

修改数据结构：

- Operation
  - 新增成员：SubAccountUin
- TaskDetail
  - 新增成员：ProcessInfo

第 45 次发布

发布时间：2023-08-29 01:14:45

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：CdcId

修改数据结构：

- InstanceInfo
  - 新增成员：HasKernelUpgrade, CdcId

第 44 次发布

发布时间：2023-07-12 01:24:00

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ModifyEsVipSecurityGroup

第 43 次发布

发布时间：2023-05-11 01:24:36

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- InstanceLog
  - 新增成员：NodeID
- KibanaView
  - 新增成员：NodeId

第 42 次发布

发布时间：2023-02-01 01:24:04

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：EnableDiagnose

第 41 次发布

发布时间：2022-12-12 01:25:49

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- IndexMetaField
  - 新增成员：IndexDocs

第 40 次发布

发布时间：2022-11-17 06:25:50

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeLogstashInstances
  - 新增入参：TagList

- RestartNodes
  - 新增入参：RestartMode, IsOffline

修改数据结构：

- LogstashInstanceInfo
  - 新增成员：CpuNum, TagList, MemSize

第 39 次发布

发布时间：2022-11-02 06:23:18

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- IndexPolicyField
  - 新增成员：ColdAction

第 38 次发布

发布时间：2022-10-25 06:33:48

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：DiskEnhance

修改数据结构：

- InstanceInfo
  - 新增成员：ProcessPercent, KibanaAlteringPublicAccess
- NodeInfo
  - 新增成员：DiskEnhance

第 37 次发布

发布时间：2022-09-28 06:30:23

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：KibanaAlteringPublicAccess
- UpgradeInstance
  - 新增入参：SkipCheckForceRestart

第 36 次发布

发布时间：2022-09-26 06:30:30

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- TaskDetail
  - 新增成员：ElapsedTime

第 35 次发布

发布时间：2022-09-16 06:24:23

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateLogstashInstance
  - 新增入参：OperationDuration
- UpdateLogstashInstance
  - 新增入参：OperationDuration

修改数据结构：

- LogstashInstanceInfo
  - 新增成员：OperationDuration

第 34 次发布

发布时间：2022-09-02 06:30:38

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：EnableHybridStorage

修改数据结构：

- InstanceInfo
  - 新增成员：EnableHybridStorage

第 33 次发布

发布时间：2022-08-16 06:24:00

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateIndex
  - 新增入参：RolloverBackingIndex

新增数据结构：

- OptionalWebServiceInfo

修改数据结构：

- InstanceInfo
  - 新增成员：OptionalWebServiceInfos, AutoIndexEnabled

第 32 次发布

发布时间：2022-08-03 06:23:52

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeLogstashInstances
  - 新增入参：VpcIds

第 31 次发布

发布时间：2022-07-14 06:10:06

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateLogstashInstance
- DeleteLogstashInstance
- DeleteLogstashPipelines
- DescribeLogstashInstanceLogs
- DescribeLogstashInstanceOperations
- DescribeLogstashInstances
- DescribeLogstashPipelines
- RestartLogstashInstance
- SaveAndDeployLogstashPipeline
- StartLogstashPipelines
- StopLogstashPipelines
- UpdateLogstashInstance
- UpdateLogstashPipelineDesc

新增数据结构：

- LogstashBindedES
- LogstashExtendedFile
- LogstashInstanceInfo
- LogstashNodeInfo
- LogstashPipeline
- LogstashPipelineInfo

第 30 次发布

发布时间：2022-07-13 06:11:03

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：OperationDuration
- DescribeInstances
  - 新增入参：VpcIds
- UpdateInstance
  - 新增入参：OperationDuration

新增数据结构：

- OperationDuration
- OperationDurationUpdated

修改数据结构：

- InstanceInfo
  - 新增成员：OperationDuration

第 29 次发布

发布时间：2022-07-05 06:13:25

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- NodeInfo
  - 新增成员：CpuNum, MemSize

第 28 次发布

发布时间：2022-06-20 06:08:20

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- InstanceInfo
  - 新增成员：EsConfigSets

第 27 次发布

发布时间：2022-05-25 06:08:55

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateIndex
- DeleteIndex
- DescribeIndexList
- DescribeIndexMeta
- UpdateIndex

修改接口：

- UpdateInstance
  - 新增入参：EsConfigSet

新增数据结构：

- BackingIndexMetaField
- EsConfigSetInfo
- IndexMetaField
- IndexOptionsField
- IndexPolicyField
- IndexSettingsField

第 26 次发布

发布时间：2022-03-21 11:04:25

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- UpdateDictionaries

第 25 次发布

发布时间：2022-03-09 08:11:10

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpgradeInstance
  - 新增入参：CosBackup

第 24 次发布

发布时间：2022-02-18 08:09:35

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- InstanceInfo
  - 新增成员：EsPrivateUrl, EsPrivateDomain

第 23 次发布

发布时间：2022-01-20 08:10:01

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- DescribeInstances
  - 新增入参：HealthStatus

修改数据结构：

- InstanceInfo
  - 新增成员：HealthStatus

第 22 次发布

发布时间：2021-12-30 08:09:09

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增出参：DealName
- DescribeInstances
  - 新增入参：ZoneList
- UpdateInstance
  - 新增出参：DealName
- UpdatePlugins
  - 新增入参：PluginType
- UpgradeLicense
  - 新增出参：DealName

修改数据结构：

- ClusterView
  - 新增成员：TotalCosStorage, SearchableSnapshotCosBucket, SearchableSnapshotCosAppId
- InstanceInfo

- 新增成员：ColdNodeType, ColdNodeNum, ColdCpuNum, ColdMemSize, ColdDiskType, ColdDiskSize, FrozenNodeType, FrozenNodeNum, FrozenCpuNum, FrozenMemSize, FrozenDiskType, FrozenDiskSize

第 21 次发布

发布时间：2021-12-22 16:41:08

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeViews

新增数据结构：

- ClusterView
- KibanaView
- NodeView

第 20 次发布

发布时间：2021-11-08 08:06:16

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：Protocol
- UpdateInstance
  - 新增入参：SwitchPrivateLink, EnableCerebro, CerebroPublicAccess, CerebroPrivateAccess

修改数据结构：

- InstanceInfo
  - 新增成员：Protocol, SecurityGroups

第 19 次发布

发布时间：2021-09-30 08:05:14

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- UpdateJdk

第 18 次发布

发布时间：2021-08-13 08:05:41

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- RestartInstance
  - 新增入参：RestartMode
- UpdateInstance
  - 新增入参：WebNodeTypeInfo

修改数据结构：

- InstanceInfo
  - 新增成员：WebNodeTypeInfo, Jdk

第 17 次发布

发布时间：2021-06-22 08:06:05

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：WebNodeTypeInfo

新增数据结构：

- KibanaNodeInfo
- WebNodeTypeInfo

修改数据结构：

- InstanceInfo
  - 新增成员：KibanaNodeInfo

第 16 次发布

发布时间：2021-05-11 08:05:19

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DiagnoseInstance
- UpdateDiagnoseSettings

第 15 次发布

发布时间：2021-03-09 08:04:52

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdatePlugins
  - 新增入参：ForceUpdate

第 14 次发布

发布时间：2020-12-28 08:04:02

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- InstanceInfo
  - 新增成员：KibanaConfig

第 13 次发布

发布时间：2020-12-25 08:04:28

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- GetRequestTargetNodeTypes
- UpdateRequestTargetNodeTypes

第 12 次发布

发布时间：2020-11-24 08:04:34

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- RestartKibana

修改接口：

- UpdateInstance
  - 新增入参：KibanaConfig

第 11 次发布

发布时间：2020-09-17 08:03:49

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- RestartNodes

修改接口：

- CreateInstance
  - 新增入参：SceneType
- UpdateInstance
  - 新增入参：MultiZoneInfo, SceneType
- UpgradeInstance
  - 新增入参：UpgradeMode

修改数据结构：

- InstanceInfo
  - 新增成员：SceneType

第 10 次发布

发布时间：2020-07-08 08:10:51

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- UpdatePlugins

修改接口：

- UpdateInstance
  - 新增入参：ScaleType

修改数据结构：

- EsDictionaryInfo
  - 新增成员：QQDict, Synonym, UpdateType
- NodeInfo
  - 新增成员：DiskEncrypt

第 9 次发布

发布时间：2020-05-08 08:10:10

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：KibanaPrivatePort

第 8 次发布

发布时间：2020-04-13 08:08:57

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- NodeInfo
  - 新增成员：DiskCount

第 7 次发布

发布时间：2020-04-09 08:08:32

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- UpdateInstance
  - 新增入参：BasicSecurityType

新增数据结构：

- LocalDiskInfo

修改数据结构：

- NodeInfo
  - 新增成员：LocalDiskInfo

第 6 次发布

发布时间：2020-01-02 10:20:54

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：NodeInfoList, TagList, BasicSecurityType
  - **修改入参：** NodeNum, NodeType, DiskSize, MultiZoneInfo
- DescribeInstances

  - 新增入参：TagList, IpList
- UpdateInstance
  - 新增入参：NodeInfoList, PublicAccess, EsPublicAcl, KibanaPublicAccess, KibanaPrivateAccess
- UpgradeInstance
  - 新增入参：BasicSecurityType
- UpgradeLicense
  - 新增入参：BasicSecurityType, ForceRestart

新增数据结构：

- EsPublicAcl
- NodeInfo
- ZoneDetail

删除数据结构：

- MultiZoneInfo

修改数据结构：

- InstanceInfo
  - 新增成员：EnableHotWarmMode, WarmNodeType, WarmNodeNum, WarmCpuNum, WarmMemSize, WarmDiskType, WarmDiskSize, NodeInfoList, EsPublicUrl, MultiZoneInfo, DeployMode, PublicAccess, EsPublicAcl, KibanaPrivateUrl, KibanaPublicAccess, KibanaPrivateAccess, SecurityType

第 5 次发布

发布时间：2019-07-18 11:17:30

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- UpgradeInstance
- UpgradeLicense

修改数据结构：

- SubTaskDetail
  - 新增成员：Type, Status, FailedIndices, FinishTime, Level

第 4 次发布

发布时间：2019-05-30 20:41:02

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：DeployMode, MultiZoneInfo, LicenseType

新增数据结构：

- MultiZoneInfo
- TagInfo

修改数据结构：

- InstanceInfo
  - 新增成员：TagList, LicenseType

第 3 次发布

发布时间：2019-03-14 16:33:24

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- DescribeInstanceLogs
- DescribeInstanceOperations

修改接口：

- CreateInstance
  - 新增入参：ClusterNameInConf
- UpdateInstance
  - 新增入参：CosBackup

新增数据结构：

- InstanceLog
- KeyValue
- Operation
- OperationDetail
- SubTaskDetail
- TaskDetail

第 2 次发布

发布时间：2019-03-07 21:24:06

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- CreateInstance
  - 新增入参：EnableDedicatedMaster, MasterNodeNum, MasterNodeType, MasterNodeDiskSize
- RestartInstance
  - 新增入参：ForceRestart
- UpdateInstance
  - 新增入参：MasterNodeNum, MasterNodeType, MasterNodeDiskSize, ForceRestart

新增数据结构：

- CosBackup

修改数据结构：

- InstanceInfo
  - 新增成员：CosBackup, AllowCosBackup
- MasterNodeInfo
  - 新增成员：MasterNodeDiskType

第 1 次发布

发布时间：2018-11-22 20:16:54

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateInstance

- DeleteInstance
- DescribeInstances
- RestartInstance
- UpdateInstance

新增数据结构：

- DictInfo
- EsAcl
- EsDictionaryInfo
- InstanceInfo
- MasterNodeInfo

简介

最近更新时间：2025-04-25 01:31:50

腾讯云 Elasticsearch Service（ES）是腾讯云基于开源搜索引擎 Elasticsearch 打造的高可用、可伸缩的云端托管 Elasticsearch 服务，部署在 VPC 内，集成了 Elasticsearch、Kibana 等组件。使用腾讯云 ES ，您可以快速部署、轻松管理、按需扩展您的 Elastic 集群，简化集群部署、管理等复杂运维操作，快速构建日志分析、异常监控、网站搜索导航、企业级搜索、BI 分析等各类业务。

API 概览

最近更新时间：2024-11-07 01:34:48

实例相关接口

接口名称	接口功能	频率限制（次/秒）
CreateIndex	创建索引	20
CreateInstance	创建ES集群实例	20
DeleteIndex	删除索引	20
DeleteInstance	销毁ES集群实例	20
DescribeDiagnose	查询智能运维诊断结果报告	20
DescribeInstanceLogs	查询ES集群日志	50
DescribeInstancePluginList	查询实例插件列表	20
DiagnoseInstance	智能运维诊断集群	20
GetDiagnoseSettings	查看智能运维配置	20
GetRequestTargetNodeTypes	获取接收客户端请求的节点类型	20
InquirePriceRenewInstance	集群续费询价	20
ModifyEsVipSecurityGroup	修改集群VIP绑定的安全组	20
RestartInstance	重启ES集群实例	20
RestartKibana	重启Kibana	20
RestartNodes	重启集群节点	20
UpdateDiagnoseSettings	更新智能运维配置	20
UpdateDictionaries	更新ES集群词典	20
UpdateIndex	更新索引	20
UpdateInstance	更新ES集群实例	20
UpdateJdk	更新实例Jdk配置	20
UpdatePlugins	变更插件列表	20
UpdateRequestTargetNodeTypes	更新接收客户端请求的节点类型	20
UpgradeInstance	升级ES集群版本	20
UpgradeLicense	升级ES商业特性	20
DescribeIndexList	获取索引列表	20
DescribeIndexMeta	获取索引元数据	20
DescribeInstanceOperations	查询实例操作记录	20
DescribeInstances	查询ES集群实例	20
DescribeViews	查询集群视图	20
CreateClusterSnapshot	创建集群快照	20
DeleteClusterSnapshot	删除快照仓库里备份的快照	20
DescribeClusterSnapshot	获取快照备份列表	20
InstallInstanceModel	安装ES集群模型	20
RestoreClusterSnapshot	快照备份恢复	20

版权所有：腾讯云计算（北京）有限责任公司

Serverless实例相关接口

接口名称	接口功能	频率限制（次/秒）
CreateServerlessSpaceV2	创建Serverless索引项目空间(new)	20
DeleteServerlessInstance	删除Serverless索引	20
UpdateServerlessSpace	更新Serverless索引空间	20
CreateServerlessInstance	创建Serverless索引	20
DeleteServerlessSpaceUser	删除Serverless空间子用户	20
DescribeServerlessSpaceUser	查看Serverless空间子用户	20
DescribeServerlessInstances	Serverless获取索引列表	20
DescribeServerlessSpaces	获取Serverless索引空间列表	20
UpdateServerlessInstance	更新Serverless索引	20
CheckMigrateIndexMetaData	检查cos迁移索引元数据	20
CreateCosMigrateToServerlessInstance	cos迁移流程	20
DescribeServerlessMetrics	获取实例对应的监控指标	100
DescribeSpaceKibanaTools	space维度的kibana获取登录token	20
DescribeUserCosSnapshotList	查询快照信息接口	20

Logstash相关接口

接口名称	接口功能	频率限制（次/秒）
CreateLogstashInstance	创建Logstash实例	20
DeleteLogstashInstance	删除Logstash实例	20
DeleteLogstashPipelines	删除Logstash管道	20
DescribeLogstashInstanceLogs	查询Logstash实例日志	20
DescribeLogstashInstances	获取Logstash实例列表	20
RestartLogstashInstance	重启Logstash实例	20
SaveAndDeployLogstashPipeline	保存并部署管道	20
StartLogstashPipelines	启动Logstash管道	20
StopLogstashPipelines	停止Logstash管道	20
UpdateLogstashInstance	更新Logstash实例	20
DescribeLogstashInstanceOperations	查询Logstash实例操作记录	20
DescribeLogstashPipelines	获取Logstash实例管道列表	20
UpdateLogstashPipelineDesc	更新管道描述	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 01:18:57

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 es.tencentcloudapi.com ，也支持指定地域域名访问，例如广州地域的域名为 es.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 es.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）	es.tencentcloudapi.com
华南地区（广州）	es.ap-guangzhou.tencentcloudapi.com
华东地区（上海）	es.ap-shanghai.tencentcloudapi.com
华东地区（南京）	es.ap-nanjing.tencentcloudapi.com
华北地区（北京）	es.ap-beijing.tencentcloudapi.com
西南地区（成都）	es.ap-chengdu.tencentcloudapi.com
西南地区（重庆）	es.ap-chongqing.tencentcloudapi.com
港澳台地区（中国香港）	es.ap-hongkong.tencentcloudapi.com
亚太东南（新加坡）	es.ap-singapore.tencentcloudapi.com
亚太东南（雅加达）	es.ap-jakarta.tencentcloudapi.com
亚太东南（曼谷）	es.ap-bangkok.tencentcloudapi.com
亚太东北（首尔）	es.ap-seoul.tencentcloudapi.com
亚太东北（东京）	es.ap-tokyo.tencentcloudapi.com
美国东部（弗吉尼亚）	es.na-ashburn.tencentcloudapi.com
美国西部（硅谷）	es.na-siliconvalley.tencentcloudapi.com
南美地区（圣保罗）	es.sa-saopaulo.tencentcloudapi.com
欧洲地区（法兰克福）	es.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）	es.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	es.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：2025-03-18 01:36:13

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 es；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 01:34:32

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 01:34:32

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 01:30:28

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 06:31:33

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。

实例相关接口
创建索引

最近更新时间：2025-04-25 01:31:46

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

创建索引

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：CreateIndex。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES集群ID 示例值：es-abcdefgh
IndexType	是	String	创建的索引类型。auto：自治索引；normal：普通索引 示例值：auto
IndexName	是	String	创建的索引名 示例值：index_name
IndexMetaJson	是	String	【必填】创建的索引元数据JSON，如mappings、settings 示例值：{"settings": {"index.number_of_shards": 1},"mappings": {"properties": {"field1": {"type": "text"}}}}
Username	否	String	集群访问用户名 示例值：elastic
Password	否	String	集群访问密码 示例值：G7j#kL9p***

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 创建自治索引

创建自治索引

输入示例

```
POST / HTTP/1.1
Host: es.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: CreateIndex
<公共请求参数>

{
"Username": "elastic",
"IndexName": "index_name",
"IndexType": "normal",
"InstanceId": "es-abcdefgh",
"IndexMetaJson": "",
"Password": "G7j#kL9p$2w"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexMetaJson	创建的索引元数据JSON IndexMetaJson的取值和预期不符。
InvalidParameter.InvalidIndexName	索引名IndexName的取值和预期不符。

错误码	描述
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

创建ES集群实例

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

创建指定规格的ES集群实例

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：CreateInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
Zone	是	String	可用区 示例值：ap-guangzhou-3
EsVersion	是	String	实例版本（支持"5.6.4"、"6.4.3"、"6.8.2"、"7.5.1"、"7.10.1"） 示例值：6.4.3
VpcId	是	String	私有网络ID 示例值：vpc-xxxxxx
SubnetId	是	String	子网ID 示例值：subnet-xxxxxx
Password	是	String	访问密码（密码需8到16位，至少包括两项（[a-z,A-Z],[0-9]和[-!@#$%&^+=_:;,.?]的特殊符号） *示例值：ascq23**cjaw*
InstanceName	否	String	实例名称（1-50 个英文、汉字、数字、连接线-或下划线_） 示例值：es_test
NodeNum	否	Integer	已废弃请使用NodeInfoList 节点数量（2-50个） 示例值：3
ChargeType	否	String	计费类型 • PREPAID：预付费，即包年包月 • POSTPAID_BY_HOUR：按小时后付费 默认值POSTPAID_BY_HOUR 示例值：POSTPAID_BY_HOUR
ChargePeriod	否	Integer	包年包月购买时长（单位由参数TimeUnit决定） 示例值：12
RenewFlag	否	String	自动续费标识 • RENEW_FLAG_AUTO：自动续费 • RENEW_FLAG_MANUAL：不自动续费，用户手动续费 ChargeType为PREPAID时需要设置，如不传递该参数，普通用户默认不自动续费，SVIP用户自动续费 示例值：RENEW_FLAG_AUTO

参数名称	必选	类型	描述
NodeType	否	String	已废弃请使用NodeInfoList 节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：ES.S1.4XLARGE64
DiskType	否	String	已废弃请使用NodeInfoList 节点磁盘类型 • CLOUD_SSD：SSD云硬盘 • CLOUD_PREMIUM：高性能云硬盘 • CLOUD_HSSD：增强型SSD云硬盘 • CLOUD_BSSD：通用型SSD云硬盘 默认值CLOUD_SSD 示例值：CLOUD_SSD
DiskSize	否	Integer	已废弃请使用NodeInfoList 节点磁盘容量（单位GB） 示例值：500
TimeUnit	否	String	计费时长单位（ChargeType为PREPAID时需要设置，默认值为“m”，表示月，当前只支持“m”） 示例值：m
AutoVoucher	否	Integer	是否自动使用代金券 • 0：不自动使用 • 1：自动使用 默认值0 示例值：0
VoucherIds.N	否	Array of String	代金券ID列表（目前仅支持指定一张代金券） 示例值：["123a**ccajoi"]
EnableDedicatedMaster	否	Boolean	已废弃请使用NodeInfoList 是否创建专用主节点 • true：开启专用主节点 • false：不开启专用主节点 默认值false 示例值：false
MasterNodeNum	否	Integer	已废弃请使用NodeInfoList 专用主节点个数（只支持3个和5个，EnableDedicatedMaster为true时该值必传） 示例值：3
MasterNodeType	否	String	已废弃请使用NodeInfoList 专用主节点类型（EnableDedicatedMaster为true时必传） • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：ES.S1.4XLARGE64
MasterNodeDiskSize	否	Integer	已废弃请使用NodeInfoList 专用主节点磁盘大小（单位GB，非必传，若传递则必须为50，暂不支持自定义） 示例值：100

参数名称	必选	类型	描述
ClusterNameInConf	否	String	集群配置文件中的ClusterName（系统默认配置为实例ID，暂不支持自定义） 示例值：testnew
DeployMode	否	Integer	集群部署方式 • 0：单可用区部署 • 1：多可用区部署，北京、上海、上海金融、广州、南京、香港、新加坡、法兰克福（白名单控制） 默认为0 示例值：0
MultiZoneInfo.N	否	Array of ZoneDetail	多可用区部署时可用区的详细信息(DeployMode为1时必传)
LicenseType	否	String	License类型 • oss：开源版 • basic：基础版 • platinum：白金版 默认值platinum 示例值：platinum
NodeInfoList.N	否	Array of NodeInfo	节点信息列表， 用于描述集群各类节点的规格信息如节点类型，节点个数，节点规格，磁盘类型，磁盘大小等
TagList.N	否	Array of TagInfo	节点标签信息列表
BasicSecurityType	否	Integer	6.8（及以上版本）基础版是否开启xpack security认证 • 1：不开启 • 2：开启 示例值：2
SceneType	否	Integer	场景化模板类型 0：不启用 1：通用 2：日志 3：搜索 示例值：0
WebNodeTypeInfo	否	WebNodeTypeInfo	可视化节点配置
Protocol	否	String	创建https集群，默认是http 示例值：https
OperationDuration	否	OperationDuration	可维护时间段
EnableHybridStorage	否	Boolean	是否开启存算分离 示例值：true
DiskEnhance	否	Integer	是否开启essd 增强型云盘 示例值：0
EnableDiagnose	否	Boolean	是否开启智能巡检 示例值：false
CdcId	否	String	cdcId，使用cdc子网时传递 示例值：cluster-xxxxxx
DisasterRecoverGroupAffinity	否	Integer	置放群组亲和度，范围[0,10]，0表示不开启 示例值：1
SubProductCode	否	String	子产品ID枚举值： 开源版："sp_es_io2"， 基础版："sp_es_basic"，白金版："sp_es_platinum"，企业版："sp_es_enterprise"，CDC白金版："sp_es_cdc_platinum"，日志增强版："sp_es_enlogging"，tsearch："sp_tsearch_io2"，logstash："sp_es_logstash"，可以为空，为空的时候后台取LicenseType映射该字段 示例值：sp_es_platinum
ReadWriteMode	否	Integer	读写分离模式：0-不开启，1-本地读写分离，2-远端读写分离 示例值：0
EnableScheduleRecoverGroup	否	Boolean	置放群组是否开启异步任务

参数名称	必选	类型	描述
			示例值：false
EnableScheduleOperationDuration	否	EnableScheduleOperationDuration	置放群组开启异步任务的可维护时间段

3. 输出参数

参数名称	类型	描述
InstanceId	String	实例ID 示例值：es-xxxxxx
DealName	String	订单号 注意：此字段可能返回 null，表示取不到有效值。 示例值：737749
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ES集群实例

根据输入参数创建ES集群实例

输入示例

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

{
"EsVersion": "6.4.3",
"VpcId": "vpc-xxxxxx",
"NodeInfoList": [
{
"NodeType": "ES.S1.SMALL2",
"NodeNum": "3",
"Type": "dedicatedMaster"
},
{
"DiskSize": "100",
"NodeType": "ES.S1.SMALL2",
"NodeNum": "2",
"Type": "hotData",
"DiskType": "CLOUD_SSD"
}
],
"Zone": "ap-guangzhou-3",
"ChargeType": "POSTPAID_BY_HOUR",
"SubnetId": "subnet-xxxxxx",
"Password": "ascq23***********cjaw",
"InstanceName": "es_test"
}
```

输出示例

```
{
"Response": {
"InstanceId": "es-abc234",
"RequestId": "ab",
"DealName": "ab"
}
}
```

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.UnAuthCreateInstance	当前用户对创建集群实例操作未授权，请添加CAM权限
AuthFailure.UnAuthorizedOperation	操作未授权。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NodeNumAndZoneError	节点数和可用区的信息不符合对应操作的规则。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidAutoVoucher	是否自动使用代金券AutoVoucher的取值和预期不符。
InvalidParameter.InvalidDeployMode	集群部署方式InvalidDeployMode的取值和预期不符。
InvalidParameter.InvalidDiskCount	节点磁盘块数DiskCount取值和预期不符。
InvalidParameter.InvalidDiskEncrypt	节点磁盘加密信息DiskEncrypt的取值与预期不符。

错误码	描述
InvalidParameter.InvalidDiskEnhance	是否启用增强型ssd云盘DiskEnhance的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidMultiZoneInfo	多可用区部署ZoneDetail的信息与预期不符。
InvalidParameter.InvalidNodeNum	节点数量NodeNum的取值和预期不符。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。
InvalidParameter.InvalidOperationDuration	可维护时间段OperationDuration的取值和预期不符。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidTagList	节点标签信息列表TagList的取值和预期不符。
InvalidParameter.InvalidType	节点类型Type的取值和预期不符。
InvalidParameter.InvalidVoucherIds	代金券ID列表VoucherIds的取值和预期不符。
InvalidParameter.InvalidZone	可用区Zone的信息与预期不符。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
InvalidParameterValue.RenewFlag	自动续费标识RenewFlag的参数取值有误。
LimitExceeded.ClusterNum	该账号下的集群数超过限额。
LimitExceeded.ResourceLimit	超出一个用户可以拥有的最大资源限制。
ResourceInUse	资源被占用。
ResourceInUse.Order	订单被锁定。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
ResourceInsufficient.Zone	可用区资源不足。
ResourceNotFound.DiskInfoNotFound	磁盘相关资源获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。
ResourceNotFound.VPCInfoNotFound	VPC资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnsupportedOperation.BasicSecurityType	不支持该认证方式。
UnsupportedOperation.LicenseError	不支持该操作，license的类型有误。
UnsupportedOperation.VPCInfoNotFound	VPC资源获取失败。

删除索引

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

删除索引

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：DeleteIndex。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES集群ID 示例值：es-abcdefgh
IndexType	是	String	删除的索引类型。auto：自治索引；normal：普通索引 示例值：auto
IndexName	是	String	删除的索引名 示例值：log_xxx
Username	否	String	集群访问用户名 示例值：elastic
Password	否	String	集群访问密码 示例值：G7j#kL9****
BackingIndexName	否	String	后备索引名 示例值：.ds-log_xxx-2024.12.16-000001

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 删除自治索引

删除自治索引

输入示例

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

```
<公共请求参数>

{
"Username": "1",
"IndexName": "1",
"IndexType": "1",
"InstanceId": "1",
"BackingIndexName": "1",
"Password": "1"
}
```

输出示例

```
{
"Response": {
"RequestId": "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
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。

错误码	描述
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

销毁ES集群实例

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

销毁集群实例

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：DeleteInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-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 删除ES集群实例

删除指定ID的集群实例

输入示例

```
https://es.tencentcloudapi.com/?Action=DeleteInstance
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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.OpenDestroyProtection	当前集群已开启销毁保护，无法直接销毁，请解除该保护后再尝试销毁
FailedOperation.RefundError	该实例不满足退费条件。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
ResourceInUse	资源被占用。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

版权所有：腾讯云计算（北京）有限责任公司

查询智能运维诊断结果报告

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询智能运维诊断结果报告

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：DescribeDiagnose。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh
Date	否	String	报告日期，格式20210301 示例值：20210301
Limit	否	Integer	报告返回份数 示例值：1

3. 输出参数

参数名称	类型	描述
Total	Integer	诊断报告个数 示例值：10
DiagnoseResults	Array of DiagnoseResult	诊断报告列表
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 查询智能运维诊断结果报告

查询智能运维诊断结果报告

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeDiagnose
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Total": 0,
"DiagnoseResults": [
{
"JobResults": [
{
"Status": 0,
"MetricDetails": [
{
"Metrics": [
{
"Dimensions": [
{
"Value": "1",
"Key": "1"
}
],
"Value": 0.0
}
],
"Key": "1"
}
],
"Detail": "1",
"Summary": "1",
"SettingDetails": [
{
"Advise": "1",
"Value": "1",
"Key": "1"
}
],
"Score": 0,
"Advise": "1",
"JobName": "1",
"LogDetails": [
{
"Count": 0,
"Advise": "1",
"Key": "1"
}
]
}
],
"InstanceId": "1",
"Completed": true,
"Score": 0,
"RequestId": "1",
"JobParam": {
"Indices": "1",
"Interval": 1,
"Jobs": [
"1"
]
},
```

```
"CreateTime": "1",
"JobType": 0
}
],
"RequestId": "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
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.DiagnoseNotFound	智能诊断相关资源未找到。
UnsupportedOperation.DiagnoseNotOpen	智能诊断未开启。

查询ES集群日志

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询用户该地域下符合条件的ES集群的日志

默认接口请求频率限制：50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：DescribeInstanceLogs。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例ID 示例值：es-xxxxxxxx
LogType	否	Integer	日志类型，默认值为1 • 1, 主日志 • 2, 搜索慢日志 • 3, 索引慢日志 • 4, GC日志 示例值：1
SearchKey	否	String	搜索词，支持LUCENE语法，如 level:WARN、ip:1.1.1.1、message:test-index等 示例值：level:INFO OR level:WARN
StartTime	否	String	日志开始时间，格式为YYYY-MM-DD HH:MM:SS, 如2019-01-22 20:15:53 示例值：2025-03-20 16:05:01
EndTime	否	String	日志结束时间，格式为YYYY-MM-DD HH:MM:SS, 如2019-01-22 20:15:53 示例值：2025-03-20 16:06:01
Offset	否	Integer	分页起始值, 默认值为0 示例值：1
Limit	否	Integer	分页大小，默认值为100，最大值100 示例值：1
OrderByType	否	Integer	时间排序方式，默认值为0 • 0, 降序 • 1, 升序 示例值：1

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	返回的日志条数 示例值：1

参数名称	类型	描述
InstanceLogList	Array of InstanceLog	日志详细信息列表 示例值：[{}]
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ES集群日志

查询实例最新的主日志

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeInstanceLogs
&InstanceId=es-f5mwm28u
&<公共请求参数>
```

输出示例


```
{
"Response": {
"TotalCount": 71633,
"InstanceLogList": [
{
"Time": "2019-01-22T10:45:36.220+08:00",
"Ip": "10.0.128.65",
"NodeID": "2131414141414",
"Level": "INFO",
"Message": "[o.e.p.o.OPackActionFilter] [1547723102001286009] forbidden request: { ID:cdcxxxabc, TYP:MainRequest, USR:nul
l, BRS:false, ACT:cluster:monitor/main, OA:10.0.128.43, IDX:, MET:GET, PTH:/, CNT:<OMITTED, LENGTH=0>, HDR:content-length,
EFF:0 } Reason: null"
},
{
"Time": "2019-01-22T10:45:35.730+08:00",
"Ip": "10.0.128.65",
"NodeID": "2131414141414",
"Level": "INFO",
"Message": "[o.e.p.o.OPackActionFilter] [1547723102001286009] forbidden request: { ID:1a8axxxabc, TYP:MainRequest, USR:nul
l, BRS:false, ACT:cluster:monitor/main, OA:10.0.128.73, IDX:, MET:GET, PTH:/, CNT:<OMITTED, LENGTH=0>, HDR:content-length,
EFF:0 } Reason: null"
}
],
"RequestId": "783d9290-dc60-4862-9340-10b632605374"
}
}
```


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.RequestTimeOut	请求超时。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidLimit	分页大小Limit的取值和预期不符。
InvalidParameter.InvalidLogType	日志类型LogType的取值和预期不符。
InvalidParameter.InvalidOffset	分页起始值Offset的取值和预期不符。
InvalidParameter.InvalidOrderByKey	排序字段OrderByKey的取值和预期不符。
InvalidParameter.InvalidQueryString	无效的lucene语法
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

版权所有：腾讯云计算（北京）有限责任公司

查询实例插件列表

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询实例插件列表

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：DescribeInstancePluginList。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
Offset	否	Integer	分页起始值, 默认值0 示例值：0
Limit	否	Integer	分页大小，默认值10 示例值：10
OrderBy	否	String	排序字段 • 1：插件名 pluginName 示例值：pluginName
OrderByType	否	String	排序方式 • 0：升序 asc • 1：降序 desc 示例值：asc
PluginType	否	Integer	0：系统插件 示例值：0

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	返回的插件个数 示例值：1
PluginList	Array of DescribeInstancePluginInfo	插件信息列表
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 查询实例插件列表

查询实例插件列表

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeInstancePluginList
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"TotalCount": 1,
"PluginList": [
{
"PluginDesc": "1",
"Status": 0,
"PluginType": 0,
"PluginVersion": "1",
"PluginUpdateTime": "1",
"PluginName": "1",
"Removable": true
}
],
"RequestId": "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
InternalError	内部错误。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameterValue.ConfigInfo	组件配置值有误。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

智能运维诊断集群

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

智能运维诊断集群

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：DiagnoseInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh
DiagnoseJobs.N	否	Array of String	需要触发的诊断项 示例值：["cluster_health"]
DiagnoseIndices	否	String	需要诊断的索引，支持通配符 示例值："*"

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 智能运维诊断集群

智能运维诊断集群

输入示例

```
https://es.tencentcloudapi.com/?Action=DiagnoseInstance
&InstanceId=ex-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "7f0e35a7-e03b-42cd-aa4f-0c1471xxxxxx"
```

```
  }
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
LimitExceeded.DiagnoseCount	超过当天智能诊断最大次数。
ResourceInUse.Diagnose	存在诊断中的作业，请等待作业诊断完成后重试。
ResourceInsufficient	资源不足。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.DiagnoseNotFound	智能诊断相关资源未找到。
UnsupportedOperation.DiagnoseJob	不支持该智能诊断任务。
UnsupportedOperation.DiagnoseNotOpen	智能诊断未开启。

查看智能运维配置

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查看智能运维配置

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：GetDiagnoseSettings。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh

3. 输出参数

参数名称	类型	描述
DiagnoseJobMetas	Array of DiagnoseJobMeta	智能运维诊断项和元信息
Status	Integer	0：开启智能运维；-1：关闭智能运维 示例值：0
CronTime	String	智能运维每天定时巡检时间 示例值：10:00:00
Count	Integer	智能运维当天已手动触发诊断次数 示例值：1
MaxCount	Integer	智能运维每天最大可手动触发次数 示例值：5
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 查看智能运维配置

查看智能运维配置

输入示例

```
https://es.tencentcloudapi.com/?Action=GetDiagnoseSettings
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Status": 0,
"Count": 0,
"DiagnoseJobMetas": [
{
"JobZhName": "1",
"JobDescription": "1",
"JobName": "1"
}
],
"CronTime": "1",
"RequestId": "1",
"MaxCount": 0
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameterValue.ConfigInfo	组件配置值有误。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

获取接收客户端请求的节点类型

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取接收客户端请求的节点类型

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：GetRequestTargetNodeTypes。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx

3. 输出参数

参数名称	类型	描述
TargetNodeTypes	Array of String	接收请求的目标节点类型列表 示例值：["hot","warm"]
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 获取接收客户端请求的节点类型

对于冷热分离的集群，可以通过该接口获取到当前集群用于接收客户端请求的节点类型

输入示例

```
https://es.tencentcloudapi.com/?Action=GetRequestTargetNodeTypes
&InstanceId=es-xxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "8fd57721-808b-4f21-8f6b-51bca79ff1xx",
"TargetNodeTypes": [
"hot",
"warm"
```

```
]
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation	操作不支持。
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。

集群续费询价

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

集群续费询价接口，续费前通过调用该接口，可获取集群续费的价格。

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：InquirePriceRenewInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例Id 示例值：es-xxxsxx

3. 输出参数

参数名称	类型	描述
OriginalPrice	Float	刊例价，即集群原始价格 示例值：100
DiscountPrice	Float	折后价 示例值：100
Discount	Float	折扣，如65折 示例值：65
Currency	String	货币，如CNY代表人民币 示例值：CNY
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 ES集群实例续费询价

ES集群实例续费询价

输入示例

```
https://es.tencentcloudapi.com/?Action=InquirePriceRenewInstance
&InstanceId=es_xdfds
&<公共请求参数>
```

输出示例

```
{
"Response": {
"Discount": 65,
"OriginalPrice": 1.15,
"DiscountPrice": 0.75,
"Currency": "CNY",
"RequestId": "027e29ad-f2b5-4929-b479-96bf541df7e7"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。

修改集群VIP绑定的安全组

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

修改绑定VIP的安全组，传安全组id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：ModifyEsVipSecurityGroup。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	es集群的实例id 示例值：es-3p7lkxxx
SecurityGroupIds.N	否	Array of String	安全组id列表 示例值：["sg-cyc7u0pp", "sg-k3jsu0jf"]

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 修改集群VIP绑定的安全组信息

修改集群VIP绑定的安全组信息

输入示例

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

{
"InstanceId": "es-3p7lkvxx",
"SecurityGroupIds": [
"sg-1",
"sg-2"
```

```
]
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidSecurityGroupIds	安全组id列表SecurityGroupIds的取值和预期不符。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.SecurityGroupNotFound	安全组信息获取失败。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

重启ES集群实例

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

重启ES集群实例(用于系统版本更新等操作)

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：RestartInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
ForceRestart	否	Boolean	是否强制重启 • true：强制重启 • false：不强制重启 默认false 示例值：false
RestartMode	否	Integer	重启模式：0 滚动重启； 1 全量重启 示例值：0
UpgradeKernel	否	Boolean	重启时选择是否升级内核patch版本 示例值：false

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 重启ES集群实例

用以进行版本更新等操作

输入示例

```
https://es.tencentcloudapi.com/?Action=RestartInstance
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "7f0e35a7-e03b-42cd-aa4f-0c1471xxxxxx"
}
}
```

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.ErrorClusterState	集群状态错误。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameterValue.ConfigInfo	组件配置值有误。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.CvmStatusStopped	有关机的cvm
UnsupportedOperation.DiskFreeSpaceInsufficient	节点磁盘可用空间不足
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。

错误码	描述
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

重启Kibana

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

重启Kibana

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：RestartKibana。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-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 重启Kibana

重启Kibana

输入示例

```
https://es.tencentcloudapi.com/?Action=RestartKibana
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "7f0e35a7-e03b-42cd-aa4f-0c1471xxxxxx"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

重启集群节点

最近更新时间：2025-05-21 01:18:56

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于重启集群节点

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：RestartNodes。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例ID 示例值：es-xxxxxxxx
NodeNames.N	是	Array of String	节点名称列表 示例值：["159229897700074xxxx\r\n"]
ForceRestart	否	Boolean	是否强制重启 示例值：true
RestartMode	否	String	可选重启模式"in-place","blue-green"，分别表示重启，蓝绿重启；默认值为"in-place" 示例值："in-place"
IsOffline	否	Boolean	节点状态，在蓝绿模式中使用；离线节点蓝绿有风险 示例值：true
CvmDelayOnlineTime	否	Integer	cvm延迟上架时间 示例值：1
ShardAllocationConcurrents	否	Integer	分片迁移并发数 示例值：2
ShardAllocationBytes	否	Integer	分片迁移并发速度 示例值：40
EnableScheduleRecoverGroup	否	Boolean	是否开启置放群组异步任务 示例值：false
EnableScheduleOperationDuration	否	EnableScheduleOperationDuration	置放群组异步任务时间段

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 重启集群节点

重启集群节点

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"NodeNames": [
"159229897700074xxxx"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.ErrorClusterState	集群状态错误。

错误码	描述
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidIpList	私有网络vip列表IpList的取值和预期不符。
InvalidParameter.InvalidNodeNames	节点名称列表NodeNames的取值和预期不符。
InvalidParameter.InvalidRestartMode	可选重启模式RestartMode的取值和预期不符。
InvalidParameter.InvalidSubnetUidList	虚拟子网络统一ID列表的取值和预期不符。
ResourceInUse	资源被占用。
ResourceInsufficient.CVM	计算资源不足，可更换可用区或调整机型。
ResourceInsufficient.SubnetIp	可用的子网IP不足。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.DiskFreeSpaceInsufficient	节点磁盘可用空间不足
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.RestartMode	离线节点蓝绿变更存在风险，请直接选择重启“in-place”。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

版权所有：腾讯云计算（北京）有限责任公司

更新智能运维配置

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新智能运维配置

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：UpdateDiagnoseSettings。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh
Status	否	Integer	0：开启智能运维；-1：关闭智能运维 示例值：0
CronTime	否	String	智能运维每天定时巡检时间，时间格式为HH:00:00，例如15:00:00 示例值：10:00:00

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 开启智能运维

开启智能运维

输入示例

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

{
"InstanceId": "es-xxxxxx",
"Status": "0"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

更新ES集群词典

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新ES集群词典

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：UpdateDictionaries。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh
IkMainDicts.N	否	Array of String	安装时填IK分词主词典COS地址，删除时填词典名如test.dic 示例值：[https://cos_address/dict.txt]
IkStopwords.N	否	Array of String	安装时填IK分词停用词词典COS地址，删除时填词典名如test.dic 示例值：[https://cos_address/dict.txt]
Synonym.N	否	Array of String	安装时填同义词词典COS地址，删除时填词典名如test.dic 示例值：[https://cos_address/dict.txt]
QQDict.N	否	Array of String	安装时填QQ分词词典COS地址，删除时填词典名如test.dic 示例值：[https://cos_address/dict.txt]
UpdateType	否	Integer	0：安装；1：删除。默认值0 示例值：0
ForceRestart	否	Boolean	是否强制重启集群。默认值false 示例值：false

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 更新ES集群词典

更新ES集群词典

输入示例

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

{
"IkStopwords": [
"stop"
],
"Synonym": [
"syno"
],
"InstanceId": "es-esadfeas",
"UpdateType": 0,
"ForceRestart": true,
"QQDict": [
"qqDict"
],
"IkMainDicts": [
"mainDict"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.EsDictionaryInfoError	由于EsDictionaryInfo错误，不允许操纵。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidUpdateType	更新类型UpdateType的取值和预期不符。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
ResourceInUse	资源被占用。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.CosInfoNotFound	COS相关信息获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

版权所有：腾讯云计算（北京）有限责任公司

更新索引

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新索引

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：UpdateIndex。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES集群ID 示例值：es-abcdefgh
IndexType	是	String	更新的索引类型。auto：自治索引；normal：普通索引 示例值：auto
IndexName	是	String	更新的索引名 示例值：index_name
UpdateMetaJson	否	String	更新的索引元数据JSON，如mappings、settings 示例值：{"index.number_of_replicas": 1}
Username	否	String	集群访问用户名 示例值：elastic
Password	否	String	集群访问密码 示例值：G7j#kL9p$2w
RolloverBackingIndex	否	Boolean	是否滚动后备索引 示例值：false

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 更新索引元数据

更新索引元数据

输入示例

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

{
"Username": "1",
"UpdateMetaJson": "1",
"IndexType": "1",
"InstanceId": "1",
"IndexName": "1",
"Password": "1"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。

错误码	描述
InvalidParameter.InvalidIndexMetaJson	创建的索引元数据JSON IndexMetaJson的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidUpdateMetaJson	更新的索引元数据json UpdateMetaJson的取值和预期不符。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

版权所有：腾讯云计算（北京）有限责任公司

更新ES集群实例

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

对集群进行节点规格变更，修改实例名称，修改配置，重置密码， 添加Kibana黑白名单等操作。参数中InstanceId为必传参数，ForceRestart为选填参数，剩余参数传递组合及含义如下：

- InstanceName：修改实例名称(仅用于标识实例)
- NodeInfoList: 修改节点配置（节点横向扩缩容，纵向扩缩容，增加主节点，增加冷节点等）
- EsConfig：修改集群配置
- Password：修改默认用户elastic的密码
- EsAcl：修改访问控制列表
- CosBackUp: 设置集群COS自动备份信息
  以上参数组合只能传递一种，多传或少传均会导致请求失败

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：UpdateInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxx
InstanceName	否	String	实例名称（1-50 个英文、汉字、数字、连接线-或下划线_） 示例值：newName
NodeNum	否	Integer	已废弃请使用NodeInfoList 节点个数（2-50个） 示例值：3
EsConfig	否	String	ES配置项（JSON格式字符串） 示例值：{"thread_pool.search.queue_size": "53400"}
Password	否	String	默认用户elastic的密码（8到16位，至少包括两项（[a-z,A-Z],[0 [-!@#$%&^*+=_:;,.?]的特殊符号） 示例值：G7j#kL9p$2w
EsAcl	否	EsAcl	可视化组件（Kibana、Cerebro）的公网访问策略 示例值：{"WhiteIpList":["123.207.123.2xx"]}
DiskSize	否	Integer	已废弃请使用NodeInfoList 磁盘大小（单位GB） 示例值：已废弃请使用NodeInfoList
NodeType	否	String	已废弃请使用NodeInfoList 节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G

参数名称	必选	类型	描述
			• ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：已废弃请使用NodeInfoList
MasterNodeNum	否	Integer	已废弃请使用NodeInfoList 专用主节点个数（只支持3个或5个） 示例值：已废弃请使用NodeInfoList
MasterNodeType	否	String	已废弃请使用NodeInfoList 专用主节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：已废弃请使用NodeInfoList
MasterNodeDiskSize	否	Integer	已废弃请使用NodeInfoList 专用主节点磁盘大小（单位GB系统默认配置为50GB,暂不支持自定 示例值：已废弃请使用NodeInfoList
ForceRestart	否	Boolean	更新配置时是否强制重启 • true强制重启 • false不强制重启 当前仅更新EsConfig时需要设置，默认值为false 示例值：已废弃请使用NodeInfoList
CosBackup	否	CosBackup	COS自动备份信息
NodeInfoList.N	否	Array of NodeInfo	节点信息列表，可以只传递要更新的节点及其对应的规格信息。支持 • 修改一种节点的个数 • 修改一种节点的节点规格及磁盘大小 • 增加一种节点类型（需要同时指定该节点的类型，个数，规格，磁 上述操作一次只能进行一种，且磁盘类型不支持修改
PublicAccess	否	String	ES集群公网访问状态 OPEN 开启 CLOSE 关闭 示例值：OPEN
EsPublicAcl	否	EsPublicAcl	公网访问控制列表
KibanaPublicAccess	否	String	Kibana公网访问状态 OPEN 开启 CLOSE 关闭 示例值：OPEN
KibanaPrivateAccess	否	String	Kibana内网访问状态 OPEN 开启 CLOSE 关闭 示例值：OPEN
BasicSecurityType	否	Integer	ES 6.8及以上版本基础版开启或关闭用户认证 示例值：1
KibanaPrivatePort	否	Integer	Kibana内网端口 示例值：5601
ScaleType	否	Integer	0: 蓝绿变更方式扩容，集群不重启（默认）1: 磁盘解挂载扩容，集 示例值：0
MultiZoneInfo.N	否	Array of ZoneDetail	多可用区部署

参数名称	必选	类型	描述
SceneType	否	Integer	场景化模板类型 -1：不启用 1：通用 2：日志 3：搜索 示例值：-1
KibanaConfig	否	String	Kibana配置项（JSON格式字符串） 示例值：{"i18n.locale":"zh-CN"}
WebNodeTypeInfo	否	WebNodeTypeInfo	可视化节点配置
SwitchPrivateLink	否	String	切换到新网络架构 示例值：switch_to_new_privatelink
EnableCerebro	否	Boolean	启用Cerebro 示例值：true
CerebroPublicAccess	否	String	Cerebro公网访问状态 OPEN 开启 CLOSE 关闭 示例值：OPEN
CerebroPrivateAccess	否	String	Cerebro内网访问状态 OPEN 开启 CLOSE 关闭 示例值：OPEN
EsConfigSet	否	EsConfigSetInfo	新增或修改的配置组信息 示例值：{"EsConfig": "{}","Type": "ldap"}
OperationDuration	否	OperationDurationUpdated	可维护时间段 示例值：{"MoreInstances":["es-8v1kjxxx","es-pa9r7xxx" [1,2,3,4,5,6,0],"TimeEnd":"06:00","TimeStart":"02:00","T
KibanaAlteringPublicAccess	否	String	是否开启Alerting 外网告警输出： OPEN 开启 CLOSE 关闭 示例值：OPEN
KibanaPrivateDomain	否	String	kibana内网自定义域名 示例值：es-xxx.internal.kibana.example.com
CerebroPrivateDomain	否	String	cerebro内网自定义域名 示例值：es-xxx.internal.cerebro.example.com
Protocol	否	String	变更为https集群，默认是http 示例值：https
OutboundPublicAcls.N	否	Array of OutboundPublicAcl	节点出站访问详细信息
OutboundPublicAccess	否	String	节点出站访问操作 OPEN 开启 CLOSE 关闭 示例值：CLOSE
CvmDelayOnlineTime	否	Integer	cvm延迟上架参数 示例值：1
ShardAllocationConcurrents	否	Integer	分片迁移并发数 示例值：2
ShardAllocationBytes	否	Integer	分片迁移并发速度 示例值：40
ReadWriteMode	否	Integer	读写分离模式：-1-不开启，1-本地读写分离，2-远端读写分离 示例值：-1
EnableScheduleRecoverGroup	否	Boolean	是否开启置放群组异步任务 示例值：false
EnableScheduleOperationDuration	否	EnableScheduleOperationDuration	置放群组异步任务可维护时间段

参数名称	必选	类型	描述
EnableDestroyProtection	否	String	开启集群保护：OPEN-开启，CLOSE-关闭 示例值：CLOSE

3. 输出参数

参数名称	类型	描述
DealName	String	订单号 注意：此字段可能返回 null，表示取不到有效值。 示例值：20250324631086496846 9xx
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Kibana密码

重置指定ES集群实例Kibana密码

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"Password": "newPwd_123"
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "1"
}
}
```

示例2 ES集群纵向扩缩容

用以对集群的节点规格（核数、内存大小）和磁盘大小进行扩缩容操作（当前仅支持纵向扩容）

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"DiskSize": "150",
```

```
"NodeType": "ES.S1.MEDIUM4"
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "1"
}
}
```

示例3 ES集群节点规格变更

用以对指定ES集群实例进行横向扩缩容，纵向扩缩容，增加专用主节点，专用主节点横向扩缩容，纵向扩缩容等操作，NodeInfoList要传递目标的全量节点信息

输入示例

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

{
"InstanceId": "es-xxxxxx",
"NodeInfoList": [
{
"NodeType": "ES.S1.SMALL2",
"NodeNum": "3",
"Type": "dedicatedMaster"
},
{
"DiskSize": "100",
"NodeType": "ES.S1.SMALL2",
"NodeNum": "2",
"Type": "hotData",
"DiskType": "CLOUD_SSD"
}
]
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "1"
}
}
```

示例4 修改ES集群实例名称

用以修改指定ES集群实例的名称

输入示例

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

{
"InstanceId": "es-xxxxxx",
"InstanceName": "newName"
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "1"
}
}
```

示例5 修改ES集群实例配置

用以对指定的ES集群实例的配置进行修改操作

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"EsConfig": "{\"action.destructive_requires_name\":\"true\"}"
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "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
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.EsDictionaryInfoError	由于EsDictionaryInfo错误，不允许操纵。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
FailedOperation.UnsupportedLocalDiskRollUpScaleUpOrDown	不支持对本地盘集群通过滚动重启方式扩缩容。
FailedOperation.UnsupportedResetNodeTypeAndScaleOutDisk	不支持在滚动重启扩容计算资源同时扩容磁盘数量。
FailedOperation.UnsupportedRestScaleDownAndModifyDisk	不支持在滚动重启缩容计算资源同时修改磁盘大小
FailedOperation.UnsupportedReverseRegulationNodeTypeAndDisk	不支持反向调节节点配置和磁盘容量。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidAutoVoucher	是否自动使用代金券AutoVoucher的取值和预期不符。
InvalidParameter.InvalidClusterName	ClusterName的取值和预期不符。
InvalidParameter.InvalidCosBackupInfo	cos自动备份信息与预期不符。
InvalidParameter.InvalidDiskCount	节点磁盘块数DiskCount取值和预期不符。
InvalidParameter.InvalidDiskEncrypt	节点磁盘加密信息DiskEncrypt的取值与预期不符。
InvalidParameter.InvalidDiskSize	节点磁盘容量DiskSize的取值和预期不符。
InvalidParameter.InvalidDiskType	节点磁盘类型DiskType的取值和预期不符。
InvalidParameter.InvalidEsACL	EsAcl的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidKibanaPrivatePort	Kibana内网访问端口的取值和预期不符。
InvalidParameter.InvalidMultiZoneInfo	多可用区部署ZoneDetail的信息与预期不符。
InvalidParameter.InvalidNodeNum	节点数量NodeNum的取值和预期不符。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameter.InvalidOperationDuration	可维护时间段OperationDuration的取值和预期不符。
InvalidParameter.InvalidPrivateAccess	内网访问PrivateAccess的取值和预期不符。

错误码	描述
InvalidParameter.InvalidPublicAccess	外网访问PublicAccess的取值和预期不符。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidRestartType	重启方式RestartType的取值和预期不符。
InvalidParameter.InvalidSubnetId	子网ID SubnetId的取值与预期不符。
InvalidParameter.InvalidType	节点类型Type的取值和预期不符。
InvalidParameter.InvalidUpdateType	更新类型UpdateType的取值和预期不符。
InvalidParameter.InvalidVoucherIds	代金券ID列表VoucherIds的取值和预期不符。
InvalidParameter.InvalidZone	可用区Zone的信息与预期不符。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.EsConfigType	ES配置类型取值有误。
InvalidParameterValue.InstanceName	实例名校验失败，按规则输入。
InvalidParameterValue.InvalidDeadline	截止日期冲突。
InvalidParameterValue.Password	密码校验不合法。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
ResourceInUse	资源被占用。
ResourceInUse.Order	订单被锁定。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.CVM	计算资源不足，可更换可用区或调整机型。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
ResourceInsufficient.SubnetIp	可用的子网IP不足。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.DiskInfoNotFound	磁盘相关资源获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。
ResourceNotFound.VPCInfoNotFound	VPC资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnsupportedOperation	操作不支持。
UnsupportedOperation.BasicSecurityType	不支持该认证方式。
UnsupportedOperation.ChangeNodeType	不支持变配节点或磁盘的配置。
UnsupportedOperation.ClusterShardNum	不支持该操作，在集群中存在分配的分片。
UnsupportedOperation.ClusterStateClose	集群中存在部分索引处于关闭状态。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.CvmStatusStopped	有关机的cvm
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程

错误码	描述
UnsupportedOperation.IndexSettingsRequireSet	集群有索引设置index.routing.allocation.require._name,or _ip, or _id
UnsupportedOperation.LicenseError	不支持该操作，license的类型有误。
UnsupportedOperation.MultiZonesUpgrade	不支持多可用区变配。
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.ScaleDownTooMuch	纵向缩容太多资源。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。
UnsupportedOperation.UpdateDiskType	不支持更新磁盘类型。
UnsupportedOperation.WebServiceType	不支持的web组件。

版权所有：腾讯云计算（北京）有限责任公司

更新实例Jdk配置

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新实例Jdk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：UpdateJdk。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES实例ID 示例值：es-abcdefgh
Jdk	否	String	Jdk类型，支持kona和oracle 示例值：oracle
Gc	否	String	Gc类型，支持g1和cms 示例值：g1
ForceRestart	否	Boolean	是否强制重启 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 UpdateJdk

更新实例Jdk类型

输入示例

```
https://es.tencentcloudapi.com/?Action=UpdateJdk
&InstanceId=es-xxxxxx
&Jdk=kona
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.GC	Gc类型的取值和预期不符。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameterValue.InvalidJDK	Jdk类型的取值和预期不符。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.OssInfoNotFound	获取oss资源失败。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

变更插件列表

最近更新时间：2025-04-25 01:31:41

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

变更插件列表

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：UpdatePlugins。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
InstallPluginList.N	否	Array of String	需要安装的插件名列表 示例值：["analysis-qq\n","sql\n"]
RemovePluginList.N	否	Array of String	需要卸载的插件名列表 示例值：["analysis-qq\n","sql\n"]
ForceRestart	否	Boolean	是否强制重启，默认值false 示例值：true
ForceUpdate	否	Boolean	是否重新安装，默认值false 示例值：true
PluginType	否	Integer	0：系统插件 示例值：0

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 批量卸载插件

批量卸载插件

输入示例

```
https://es.tencentcloudapi.com/?Action=UpdatePlugins
&InstanceId=es-xxxxxxxx
&RemovePluginList.0=analysis-qq
```

```
&RemovePluginList.1=sql
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

示例2 批量安装插件

批量安装插件

输入示例

```
https://es.tencentcloudapi.com/?Action=UpdatePlugins
&InstanceId=es-xxxxxxxx
&InstallPluginList.0=analysis-qq
&InstallPluginList.1=sql
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.FileNameError	文件名不符合规范。
FailedOperation.FileSizeError	文件大小不符合要求。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.InstallPluginList	为集群稳定起见，当前只支持同时安装1个用户插件。
InvalidParameterValue.PluginType	插件类型PluginType的取值有误。
LimitExceeded.PluginInstall	超过最大自定义插件数量，请先删除部分自定义插件。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.CosInfoNotFound	COS相关信息获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

版权所有：腾讯云计算（北京）有限责任公司

更新接收客户端请求的节点类型

最近更新时间：2025-04-25 01:31:41

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新接收客户端请求的节点类型

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：UpdateRequestTargetNodeTypes。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值："es-xxxxxxxx"
TargetNodeTypes.N	是	Array of String	接收请求的目标节点类型列表 示例值：["hot"]

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 更新接收客户端请求的节点类型

对于冷热分离的集群，在温节点节点规格比较小的情况下，可以把来自客户端的请求只发送到热节点上，只让热节点承担协调节点的功能。

输入示例

```
https://es.tencentcloudapi.com/?Action=UpdateRequestTargetNodeTypes
&InstanceId=es-xxxxxx
&TargetNodeTypes.0=hot
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "8fd57721-808b-4f21-8f6b-51bca79ff1xx"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation	操作不支持。
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

升级ES集群版本

最近更新时间：2025-04-25 01:31:41

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

升级ES集群版本

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：UpgradeInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
EsVersion	是	String	目标ES版本，支持：” 6.4.3 “, "6.8.2"，"7.5.1", "7.10.1", "7.14.2" 示例值：6.4.3
CheckOnly	否	Boolean	是否只做升级检查，默认值为false 示例值：false
LicenseType	否	String	目标商业特性版本： • oss 开源版 • basic 基础版 当前仅在5.6.4升级6.x版本时使用，默认值为basic 示例值：basic
BasicSecurityType	否	Integer	6.8（及以上版本）基础版是否开启xpack security认证 • 1：不开启 • 2：开启 示例值：security
UpgradeMode	否	String	升级方式： • scale 蓝绿变更 • restart 滚动重启 默认值为scale 示例值：scale
CosBackup	否	Boolean	升级版本前是否对集群进行备份，默认不备份 示例值：false
SkipCheckForceRestart	否	Boolean	滚动模式时，是否跳过检查，进行强制重启。默认值为false 示例值：false
CvmDelayOnlineTime	否	Integer	cvm延迟上架参数 示例值：1
ShardAllocationConcurrents	否	Integer	分片迁移并发数 示例值：2

参数名称	必选	类型	描述
ShardAllocationBytes	否	Integer	分片迁移并发速度 示例值：40
EnableScheduleRecoverGroup	否	Boolean	是否开启置放群组异步任务 示例值：false
EnableScheduleOperationDuration	否	EnableScheduleOperationDuration	置放群组异步任务时间段

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 升级ES集群版本

升级ES集群版本

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"EsVersion": "6.4.3"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.ErrorClusterState	集群状态错误。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.UpgradeMode	升级方式UpgradeMode的取值有误。
LimitExceeded.NodeNumOrIndices	节点数量或索引存储量超过限额。
LimitExceeded.ResourceLimit	超出一个用户可以拥有的最大资源限制。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.SubnetIp	可用的子网IP不足。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnsupportedOperation.ClusterStateClose	集群中存在部分索引处于关闭状态。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.CosBackUp	该版本不支持cos备份。集群需要重启来允许cos备份。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.IndexSettingsRequireSet	集群有索引设置index.routing.allocation.require._name,or _ip, or _id
UnsupportedOperation.LicenseError	不支持该操作，license的类型有误。
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

升级ES商业特性

最近更新时间：2025-04-25 01:31:41

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

升级ES商业特性

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：UpgradeLicense。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
LicenseType	是	String	License类型 • oss：开源版 • basic：基础版 • platinum：白金版 默认值platinum 示例值：platinum
AutoVoucher	否	Integer	是否自动使用代金券 • 0：不自动使用 • 1：自动使用 默认值0 示例值：1
VoucherIds.N	否	Array of String	代金券ID列表（目前仅支持指定一张代金券） 示例值：12231
BasicSecurityType	否	Integer	6.8（及以上版本）基础版是否开启xpack security认证 • 1：不开启 • 2：开启 • 不传参时会默认维持原状，传参时需注意只能由不开启变为开启，无法由开启变为不开启 示例值：0
ForceRestart	否	Boolean	是否强制重启 • true强制重启 • false不强制重启 默认值false 示例值：TRUE

3. 输出参数

参数名称	类型	描述
DealName	String	订单号 注意：此字段可能返回 null，表示取不到有效值。

参数名称	类型	描述
		示例值：2023123321
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 升级ES商业特性

升级ES商业特性

输入示例

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

{
"InstanceId": "es-xxxxxxxx",
"LicenseType": "platinum"
}
```

输出示例

```
{
"Response": {
"RequestId": "1",
"DealName": "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
FailedOperation.EsDictionaryInfoError	由于EsDictionaryInfo错误，不允许操纵。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidAutoVoucher	是否自动使用代金券AutoVoucher的取值和预期不符。
InvalidParameter.InvalidCosBackupInfo	cos自动备份信息与预期不符。
InvalidParameter.InvalidDiskCount	节点磁盘块数DiskCount取值和预期不符。
InvalidParameter.InvalidDiskSize	节点磁盘容量DiskSize的取值和预期不符。
InvalidParameter.InvalidDiskType	节点磁盘类型DiskType的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidKibanaPrivatePort	Kibana内网访问端口的取值和预期不符。
InvalidParameter.InvalidNodeNum	节点数量NodeNum的取值和预期不符。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。
InvalidParameter.InvalidPrivateAccess	内网访问PrivateAccess的取值和预期不符。
InvalidParameter.InvalidRestartType	重启方式RestartType的取值和预期不符。
InvalidParameter.InvalidType	节点类型Type的取值和预期不符。
InvalidParameter.InvalidVoucherIds	代金券ID列表VoucherIds的取值和预期不符。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.EsConfigType	ES配置类型取值有误。
InvalidParameterValue.InstanceName	实例名校验失败，按规则输入。
InvalidParameterValue.InvalidDeadline	截止日期冲突。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
ResourceInUse	资源被占用。
ResourceInUse.Order	订单被锁定。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DataNodeNotFound	数据节点信息未找到。
ResourceNotFound.DiskInfoNotFound	磁盘相关资源获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。

错误码	描述
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnsupportedOperation	操作不支持。
UnsupportedOperation.BasicSecurityType	不支持该认证方式。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.LicenseError	不支持该操作，license的类型有误。
UnsupportedOperation.LocalDisk	不支持本地盘机型。
UnsupportedOperation.MultiZonesUpgrade	不支持多可用区变配。
UnsupportedOperation.Plugin	不支持该插件。

版权所有：腾讯云计算（北京）有限责任公司

获取索引列表

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取索引列表

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：DescribeIndexList。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
IndexType	是	String	索引类型。auto：自治索引；normal：普通索引 示例值：auto
InstanceId	否	String	ES集群ID 示例值：es-abcdefgh
IndexName	否	String	索引名，若填空则获取所有索引 示例值：log_xxx
Username	否	String	集群访问用户名 示例值：elastic
Password	否	String	集群访问密码 示例值：ascq23***cjaw
Offset	否	Integer	分页起始位置 示例值：0
Limit	否	Integer	一页展示数量 示例值：20
OrderBy	否	String	排序字段，支持索引名：IndexName、索引存储量：IndexStorage、索引创建时间：IndexCreateTime 示例值：storage
IndexStatusList.N	否	Array of String	过滤索引状态 示例值：["GREEN"]
Order	否	String	排序顺序，支持asc、desc，默认为desc 数据格式"asc","desc" 示例值：asc

3. 输出参数

参数名称	类型	描述
IndexMetaFields	Array of IndexMetaField	索引元数据字段 注意：此字段可能返回 null，表示取不到有效值。
TotalCount	Integer	查询总数 注意：此字段可能返回 null，表示取不到有效值。

参数名称	类型	描述
		示例值：10
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 获取索引元数据

获取索引元数据

输入示例

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

{
"Username": "elastic",
"IndexName": "index_test",
"IndexType": "auto",
"InstanceId": "es-abcdefgh",
"Limit": 0,
"Offset": 10,
"Password": "ascq23***********cjaw"
}
```

输出示例

```
{
"Response": {
"IndexMetaFields": [
{
"IndexType": "auto",
"IndexName": "index_test",
"IndexMetaJson": "{\"f1\":52}",
"IndexStatus": "green",
"IndexStorage": 0,
"IndexCreateTime": "2024-12-05 17:22:44",
"BackingIndices": [
{
"IndexName": "index_test0",
"IndexStatus": "green",
"IndexStorage": 0,
"IndexPhrase": "data_hot",
"IndexCreateTime": "2024-12-05 17:22:44"
}
],
"ClusterId": "es-hhio223",
"ClusterName": "test-cluster",
"ClusterVersion": "7.14.0",
"IndexPolicyField": {
"WarmEnable": "true",
"WarmMinAge": "16",
```

```
"ColdEnable": "true",
"ColdMinAge": "8",
"FrozenEnable": "false",
"FrozenMinAge": "0",
"ColdAction": "migrate"
},
"IndexOptionsField": {
"ExpireMaxAge": "1d",
"ExpireMaxSize": "1024",
"RolloverMaxAge": "1d",
"RolloverDynamic": "true",
"ShardNumDynamic": "true",
"TimestampField": "TIMESTAMP",
"WriteMode": "append_only"
},
"IndexSettingsField": {
"NumberOfShards": "2",
"NumberOfReplicas": "1",
"RefreshInterval": "30s"
},
"AppId": 1,
"IndexDocs": 1
}
],
"TotalCount": 0,
"RequestId": "ah782-j*************c-ih89"
}
}
```


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.RequestTimeOut	请求超时。
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

版权所有：腾讯云计算（北京）有限责任公司

获取索引元数据

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取索引元数据

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：DescribeIndexMeta。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	ES集群ID 示例值：es-abcdefgh
IndexType	是	String	索引类型。auto：自治索引；normal：普通索引 示例值：auto
IndexName	是	String	索引名，若填空则获取所有索引 示例值：log_xxx
Username	否	String	集群访问用户名 示例值：elastic
Password	否	String	集群访问密码 示例值：G7j#kL9p$2w

3. 输出参数

参数名称	类型	描述
IndexMetaField	IndexMetaField	索引元数据字段 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 获取索引元数据

获取索引元数据

输入示例

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

```
<公共请求参数>

{
"InstanceId": "es-ikbe4ku1",
"Username": "elastic",
"Password": "G7j#kL9p$2w",
"IndexName": "log_xxx",
"IndexType": "auto"
}
```

输出示例


```
{
"Response": {
"IndexMetaField": {
"AppId": 1256050235,
"ClusterId": "es-ikbe4ku1",
"ClusterName": "AT-ES索引测试",
"ClusterVersion": "7.14.2",
"IndexType": "normal",
"IndexName": "at-20250306061507",
"IndexMetaJson": "{\"mappings\":{\"properties\":{\"at_field_20250306062058\":{\"analyzer\":\"ik_max_word\",\"type\":\"text
\"},\"charge\":{\"type\":\"keyword\"},\"codea\":{\"type\":\"keyword\"},\"lat\":{\"type\":\"keyword\"},\"level\":{\"type
\":\"keyword\"},\"location\":{\"type\":\"keyword\"},\"log\":{\"type\":\"keyword\"},\"name\":{\"type\":\"keyword\"},\"prov
\":{\"type\":\"keyword\"},\"remark\":{\"type\":\"keyword\"}}},\"settings\":{\"index.number_of_replicas\":\"2\",\"index.num
ber_of_shards\":\"1\",\"index.refresh_interval\":\"1m\"}}",
"IndexStatus": "green",
"IndexStorage": 366412,
"IndexCreateTime": "2025-03-06 06:15:07",
"IndexDocs": 348,
"IndexSettingsField": {
"NumberOfShards": "1",
"NumberOfReplicas": "2",
"RefreshInterval": "1m"
},
"IndexAliasesField": [],
"IndexPolicyField": {
"WarmEnable": "true",
"WarmMinAge": "16",
"ColdEnable": "true",
"ColdMinAge": "8",
"FrozenEnable": "false",
"FrozenMinAge": "0",
"ColdAction": "migrate"
},
"IndexOptionsField": {
"ExpireMaxAge": "1d",
"ExpireMaxSize": "1024",
"RolloverMaxAge": "1d",
"RolloverDynamic": "true",
"ShardNumDynamic": "true",
"TimestampField": "TIMESTAMP",
"WriteMode": "append_only"
},
"BackingIndices": [
{
```

```
"IndexName": "index_test0",
"IndexStatus": "green",
"IndexStorage": 0,
"IndexPhrase": "data_hot",
"IndexCreateTime": "2024-12-05 17:22:44"
}
]
},
"RequestId": "8b051ef0-5802-49cd-a73f-efe0e68662cf"
}
}
```

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.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexName	索引名IndexName的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

查询实例操作记录

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询实例指定条件下的操作记录

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：DescribeInstanceOperations。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例ID 示例值：es-xxxxxxxx
StartTime	是	String	起始时间, e.g. "2019-03-07 16:30:39" 示例值：2018-01-01 00:00:00
EndTime	是	String	结束时间, e.g. "2019-03-30 20:18:03" 示例值：2025-03-20 23:59:59
Offset	是	Integer	分页起始值 示例值：0
Limit	是	Integer	分页大小 示例值：1

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	操作记录总数 示例值：24
Operations	Array of Operation	操作记录 示例值：[{}]
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ES集群实例操作记录

查询ES集群实例操作记录

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeInstanceOperations
&InstanceId=es-f5mwm28u
&StartTime=2019-01-30 20:18:03
&EndTime=2019-03-30 20:18:03
&Offset=0
&Limit=30
&<公共请求参数>
```

输出示例


```
{
"Response": {
"TotalCount": 1,
"Operations": [
{
"Id": 6173,
"StartTime": "2019-03-07 16:30:39",
"Type": "CreateInstance",
"Detail": {
"OldInfo": [],
"NewInfo": []
},
"Result": "completed",
"Tasks": [
{
"Name": "prepareResource",
"Progress": 1,
"FinishTime": "2019-03-07 16:31:11",
"SubTasks": [],
"ElapsedTime": 0
},
{
"Name": "deployESCluster",
"Progress": 1,
"FinishTime": "2019-03-07 16:34:32",
"SubTasks": [],
"ElapsedTime": 0
},
{
"Name": "deployKibana",
"Progress": 1,
"FinishTime": "2019-03-07 16:35:13",
"SubTasks": [],
"ElapsedTime": 0
},
{
"Name": "configLB",
"Progress": 1,
"FinishTime": "2019-03-07 16:35:15",
"SubTasks": [],
"ElapsedTime": 0
}
],
"Progress": 1
}
],
```

```
"RequestId": "870dd618-b1ae-40cc-a5a9-22b867367ed7"
}
}
```

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.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidLimit	分页大小Limit的取值和预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

查询ES集群实例

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询用户该地域下符合条件的所有实例

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：DescribeInstances。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
Zone	否	String	集群实例所属可用区，不传则默认所有可用区 示例值：ap-guangzhou-3
InstanceIds.N	否	Array of String	集群实例ID列表 示例值：es-abcdefgh
InstanceNames.N	否	Array of String	集群实例名称列表 示例值：testnew
Offset	否	Integer	分页起始值, 默认值0 示例值：0
Limit	否	Integer	分页大小，默认值20 示例值：10
OrderByKey	否	Integer	排序字段 • 1：实例ID • 2：实例名称 • 3：可用区 • 4：创建时间 若orderByKey未传递则按创建时间降序排序 示例值：1
OrderByType	否	Integer	排序方式 • 0：升序 • 1：降序 若传递了orderByKey未传递orderByType, 则默认升序 示例值：0
TagList.N	否	Array of TagInfo	节点标签信息列表
IpList.N	否	Array of String	私有网络vip列表 示例值：10.2.3.4
ZoneList.N	否	Array of String	可用区列表 示例值：ap-guangzhou-3
HealthStatus.N	否	Array of Integer	健康状态筛列表:0表示绿色，1表示黄色，2表示红色,-1表示未知 示例值：0

参数名称	必选	类型	描述
VpcIds.N	否	Array of String	Vpc列表 筛选项 示例值：vpc-gz5fa86l
CdcId	否	String	cdc集群id 示例值：cluster-262n63e8

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	返回的实例个数 示例值：2
InstanceList	Array of InstanceInfo	实例详细信息列表 示例值：[{}]
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ES集群实例

查询ES集群实例

输入示例

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

{}
```

输出示例

```
{
"Response": {
"TotalCount": 1,
"InstanceList": [
{
"InstanceId": "es-7sy7efoi",
"InstanceName": "_some_name_",
"InstanceType": 2,
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-3",
"AppId": 130000000,
"Uin": "100000000",
"VpcUid": "vpc-xxxxxx",
"SubnetUid": "subnet-xxxxxx",
"CdcId": "cluster-xxxxxxxx",
"Status": 1,
"ChargeType": "POSTPAID_BY_HOUR",
"ChargePeriod": 0,
"RenewFlag": "",
"NodeInfoList": [
{
```

```
"Type": "hotData",
"NodeNum": 3,
"NodeType": "ES.SA2.MEDIUM4",
"CpuNum": 0,
"MemSize": 0,
"DiskType": "CLOUD_SSD",
"DiskSize": 20,
"DiskEnhance": 0,
"DiskEncrypt": 0,
"DiskCount": 1,
"LocalDiskInfo": null
}
],
"NodeType": "ES.SA2.MEDIUM4",
"NodeNum": 3,
"CpuNum": 2,
"MemSize": 4,
"DiskType": "CLOUD_SSD",
"AutoIndexEnabled": true,
"DiskSize": 20,
"EnableHotWarmMode": false,
"WarmNodeType": "",
"WarmNodeNum": 0,
"WarmCpuNum": 0,
"WarmMemSize": 0,
"WarmDiskType": "",
"WarmDiskSize": 0,
"ColdNodeType": "",
"ColdNodeNum": 0,
"ColdCpuNum": 0,
"ColdMemSize": 0,
"ColdDiskType": "",
"ColdDiskSize": 0,
"FrozenNodeType": "",
"FrozenNodeNum": 0,
"FrozenCpuNum": 0,
"FrozenMemSize": 0,
"FrozenDiskType": "",
"FrozenDiskSize": 0,
"MasterNodeInfo": {
"EnableDedicatedMaster": false,
"MasterNodeType": "",
"MasterNodeNum": 0,
"MasterNodeCpuNum": 0,
"MasterNodeMemSize": 0,
"MasterNodeDiskType": "",
"MasterNodeDiskSize": 0
},
"KibanaNodeInfo": {
"KibanaNodeType": "ES.SA2.SMALL2",
"KibanaNodeNum": 1,
"KibanaNodeCpuNum": 1,
"KibanaNodeMemSize": 2,
"KibanaNodeDiskType": "CLOUD_PREMIUM",
"KibanaNodeDiskSize": 50
},
"EsDomain": "es-7sy7efoi.myelasticsearch.com",
```

```
"EsPrivateDomain": "es-7sy7efoi-esinternal.kibana.myelasticsearch.com",
"EsVip": "{vip}",
"EsPort": 9200,
"KibanaUrl": "https://es-7sy7efoi.kibana.myelasticsearch.com:5601",
"KibanaPrivateUrl": "",
"PublicAccess": "CLOSE",
"KibanaPublicAccess": "OPEN",
"KibanaPrivateAccess": "CLOSE",
"KibanaAlteringPublicAccess": "CLOSE",
"Protocol": "",
"SecurityGroups": [],
"EsPublicUrl": "",
"EsPrivateUrl": "https://es-7sy7efoi-esinternal.kibana.myelasticsearch.com:9200",
"EsVersion": "7.14.2",
"EsConfig": "{\"http.cors.allow-methods\":\"OPTIONS,HEAD,GET,POST,PUT,DELETE\"}",
"EsConfigSets": [],
"KibanaConfig": "",
"EsAcl": {
"WhiteIpList": [
"127.0.0.1"
],
"BlackIpList": []
},
"EsPublicAcl": {
"WhiteIpList": [],
"BlackIpList": []
},
"IkConfig": {
"UpdateType": "",
"MainDict": [],
"Stopwords": [],
"Synonym": [],
"QQDict": []
},
"AllowCosBackup": true,
"CosBackup": {
"IsAutoBackup": false,
"BackupTime": ""
},
"DeployMode": 0,
"MultiZoneInfo": [
{
"Zone": "ap-guangzhou-3",
"SubnetId": "subnet-q29llw8g"
}
],
"TagList": [],
"HealthStatus": 0,
"SecurityType": 2,
"LicenseType": "platinum",
"SceneType": 1,
"Jdk": "kona",
"WebNodeTypeInfo": {
"NodeNum": 1,
"NodeType": "ES.SA2.SMALL2"
},
"CreateTime": "2022-07-06 20:52:48",
```

```
"UpdateTime": "2022-07-07 17:35:29",
"Deadline": "2017-12-04 00:00:00",
"OperationDuration": null,
"OptionalWebServiceInfos": [
{
"Status": 0,
"PrivateAccess": "CLOSE",
"PublicAccess": "CLOSE",
"PrivateUrl": "http://es-xxxxxxxx.internal.cerebro.tencentelasticsearch.com:9000",
"Version": "0.9.4",
"Type": "cerebro",
"PublicUrl": "https://es-xxxxxxxx.cerebro.tencentelasticsearch.com:9000"
}
],
"ProcessPercent": 0.5,
"EnableHybridStorage": true,
"HasKernelUpgrade": true,
"SubProductCode": "sp_es_platium"
}
],
"RequestId": "c1689946-7206-447c-9ee7-33b1cc0360a1"
}
}
```


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.UnAuthDescribeInstances	当前用户对查询集群实例操作未授权，请添加CAM权限。
AuthFailure.UnAuthorizedOperation	操作未授权。
FailedOperation.GetTagInfoError	获取实例的标签列表错误。

错误码	描述
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidIp	Ip地址取值和预期不符。
InvalidParameter.InvalidIpList	私有网络vip列表IpList的取值和预期不符。
InvalidParameter.InvalidLimit	分页大小Limit的取值和预期不符。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameter.InvalidOffset	分页起始值Offset的取值和预期不符。
InvalidParameter.InvalidOrderByKey	排序字段OrderByKey的取值和预期不符。
InvalidParameter.InvalidOrderByType	排序方式OrderByType的取值和预期不符。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidTagInfo	节点标签信息TagInfo的取值和预期不符。
InvalidParameter.InvalidTagList	节点标签信息列表TagList的取值和预期不符。
InvalidParameter.InvalidZone	可用区Zone的信息与预期不符。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

版权所有：腾讯云计算（北京）有限责任公司

查询集群视图

最近更新时间：2025-04-25 01:31:44

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询集群各视图数据，包括集群维度、节点维度、Kibana维度

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：DescribeViews。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例ID 示例值：es-xxxxxxxx

3. 输出参数

参数名称	类型	描述
ClusterView	ClusterView	集群维度视图 注意：此字段可能返回 null，表示取不到有效值。 示例值：ClusterView
NodesView	Array of NodeView	节点维度视图 注意：此字段可能返回 null，表示取不到有效值。 示例值：NodesView
KibanasView	Array of KibanaView	Kibana维度视图 注意：此字段可能返回 null，表示取不到有效值。 示例值：KibanasView
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 查询集群视图

查询集群视图

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeViews
&InstanceId=es-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6127baed-bc4c-4c0d-b52d-6463a722795b",
"ClusterView": {
"Health": 0,
"NodeNum": 3,
"TotalNodeNum": 3,
"DataNodeNum": 3,
"IndexNum": 121,
"DocNum": 68195345,
"DiskUsedInBytes": 27909384217,
"ShardNum": 1033,
"PrimaryShardNum": 516,
"RelocatingShardNum": 0,
"InitializingShardNum": 0,
"UnassignedShardNum": 0,
"Visible": 1,
"Break": 0,
"AvgDiskUsage": 0.075802238966579,
"AvgMemUsage": 0.82355177962355,
"AvgCpuUsage": 0.0097774027927309,
"TotalDiskSize": 600,
"TotalCosStorage": 0,
"SearchableSnapshotCosBucket": "",
"SearchableSnapshotCosAppId": "",
"TargetNodeTypes": [
"data"
]
},
"NodesView": [
{
"NodeId": "1739331190000739932",
"NodeIp": "9.29.48.61",
"NodeHttpIp": "9.29.48.61",
"Zone": "ap-hongkong-3",
"Hidden": false,
"NodeRole": "hotData",
"Visible": 1,
"Break": 0,
"DiskSize": 200,
"DiskUsage": 0.075739699850416,
"MemSize": 8,
"MemUsage": 0.82319480451571,
"JvmMemUsage": 0.6,
"CpuNum": 4,
"CpuUsage": 0.0048451231329569,
"ShardNum": 345,
"DiskIds": [
"disk-6eur8xxx"
],
"IsCoordinationNode": true,
"CVMStatus": "RUNNING",
"CVMDisasterRecoverGroupId": "",
"CVMDisasterRecoverGroupStatus": 0
},
{
```

```
"NodeId": "1739331190000740032",
"NodeIp": "9.29.50.27",
"NodeHttpIp": "9.29.50.27",
"Zone": "ap-hongkong-3",
"Hidden": false,
"NodeRole": "hotData",
"Visible": 1,
"Break": 0,
"DiskSize": 200,
"DiskUsage": 0.076343043587719,
"MemSize": 8,
"MemUsage": 0.82361871962329,
"JvmMemUsage": 0.34,
"CpuNum": 4,
"CpuUsage": 0.0048869730840559,
"ShardNum": 344,
"DiskIds": [
"disk-0pwecycm"
],
"IsCoordinationNode": true,
"CVMStatus": "RUNNING",
"CVMDisasterRecoverGroupId": "",
"CVMDisasterRecoverGroupStatus": 0
},
{
"NodeId": "1739331190000739832",
"NodeIp": "9.29.51.17",
"NodeHttpIp": "9.29.51.17",
"Zone": "ap-hongkong-3",
"Hidden": false,
"NodeRole": "hotData",
"Visible": 1,
"Break": 0,
"DiskSize": 200,
"DiskUsage": 0.07533630210082,
"MemSize": 8,
"MemUsage": 0.8215642493271,
"JvmMemUsage": 0.62,
"CpuNum": 4,
"CpuUsage": 0.0041350619841615,
"ShardNum": 344,
"DiskIds": [
"disk-2xtqdaju"
],
"IsCoordinationNode": true,
"CVMStatus": "RUNNING",
"CVMDisasterRecoverGroupId": "",
"CVMDisasterRecoverGroupStatus": 0
}
],
"KibanasView": [
{
"Ip": "9.29.49.38",
"NodeId": "1739331190000739832",
"Zone": "ap-hongkong-3",
"DiskSize": 50,
"DiskUsage": 0.19899999999999998,
```

```
"MemSize": 2,
"MemUsage": 0.76055485347424,
"CpuNum": 1,
"CpuUsage": 0.029833333333333
}
]
}
}
```

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.UnAuthCreateInstance	当前用户对创建集群实例操作未授权，请添加CAM权限
InternalError	内部错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceInUse	资源被占用。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。

安装ES集群模型

最近更新时间：2025-04-25 01:31:43

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

ES集群安装模型接口

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：InstallInstanceModel。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
UsrCosModelUrlList.N	否	Array of String	客户上传到cos的模型地址，单次请求限制一个。cos文件为压缩文件，格式包括：zip、tgz和tar.gz 示例值：["https://***.myqcloud.com/**.zip"]
ModelNames.N	否	Array of String	客户指定安装的模型名称，可为空，默认为模型文件名 示例值：[**]
TaskTypes.N	否	Array of String	模型使用的任务类型，包括：fill_mask, ner, question_answering, text_classification, text_embedding, text_expansion, text_similarity和zero_shot_classification，默认为text_embedding 示例值：["text_embedding"]

3. 输出参数

参数名称	类型	描述
FlowId	String	发起异步流程的flowId 示例值：1111111
ErrMsg	String	调用接口的错误信息 示例值：The license of basic is basic. Don't supported to install model.
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 安装模型调用示例

输入示例

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

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

{
"InstanceId": "es-xxxxxxxx",
"UsrCosModelUrlList": [
"https://modeltestxxxx-sh-12560502xx.cos.ap-shanghai.myqcloud.com/bge-base-zh-v1.5.zip"
],
"ModelNames": [
"wit-test"
],
"TaskTypes": [
"text_embedding"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "d6c1d15f-5eab-11ef-9b1d-525400a23f29",
"FlowId": "17279"
}
}
```

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.InvalidEsVersion	实例版本EsVersion的取值和预期不符。

错误码	描述
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。

快照备份恢复

最近更新时间：2025-04-25 01:31:42

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

快照备份恢复，从仓库中恢复快照到集群中

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：RestoreClusterSnapshot。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例Id，格式：es-xxxx 示例值：es-xxx
RepositoryName	是	String	仓库名称 示例值：ES_AUTO_BACKUP
SnapshotName	是	String	集群快照名称 示例值：test_snapshot
TargetInstanceId	是	String	目标集群实例Id，格式：es-xxxx，如果是恢复到本地，则和InstanceId一致 示例值：es-xxxxxxxx
Password	是	String	elastic用户名对应的密码信息 示例值：G7j#kL9p$2w
IndexSettings	否	String	要在所有恢复的索引中添加或更改的设置的逗号分隔列表。使用此参数可以在恢复快照时覆盖索引设置。 示例值：{"index.number_of_replicas": 1}
IncludeGlobalState.N	否	Array of String	不应从快照还原的以逗号分隔的索引设置列表。 示例值：["false"]
Indices	否	String	需要恢复的索引名称，非必填，为空则表示恢复所有 支持传多个索引名称 示例值：,-.
Partial	否	String	如果为 false，则如果快照中包含的一个或多个索引没有所有主分片可用，则整个恢复操作将失败。默认为 false， 如果为 true，则允许恢复具有不可用分片的索引的部分快照。只有成功包含在快照中的分片才会被恢复。所有丢失的碎片将被重新创建为空 示例值：true

3. 输出参数

参数名称	类型	描述
InstanceId	String	集群实例id 示例值：es-xxx
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

参数名称	类型	描述
		要提供该次请求的 RequestId。

4. 示例

示例1 RestoreClusterSnapshot 恢复快照数据到集群

输入示例

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

{
"InstanceId": "es-qpzizjym",
"Password": "pass",
"RepositoryName": "ES_AUTO_BACKUP",
"SnapshotName": "test1",
"Indices": "testnew",
"Partial": "true",
"TargetInstanceId": "es-qpzizjym"
}
```

输出示例

```
{
"Response": {
"InstanceId": "es-qpzizjym",
"RequestId": "test"
}
}
```

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.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameterValue.Password	密码校验不合法。
ResourceNotFound.CosInfoNotFound	COS相关信息获取失败。

获取快照备份列表

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取快照备份列表

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：DescribeClusterSnapshot。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例Id，格式：es-xxxx 示例值：es-xxxxxxxx
RepositoryName	否	String	快照仓库名称 示例值：BACKUP
SnapshotName	否	String	集群快照名称 示例值：test_snapshot

3. 输出参数

参数名称	类型	描述
InstanceId	String	集群实例Id，格式：es-xxxx 示例值：es-xxxxxxxx
Snapshots	Array of Snapshots	快照备份详情列表 示例值：返回[]
RepositoryName	String	快照仓库名称 示例值：BACKUP
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 DescribeClusterSnapshot 查询快照列表

输入示例

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

```
{
"InstanceId": "es-f1m2enyd",
"RepositoryName": "ES_AUTO_BACKUP",
"SnapshotName": "snapshot_test"
}
```

输出示例

```
{
"Response": {
"RequestId": "50df75f7-ec95-4c32-b7b0-a8e98ae74fef11",
"RepositoryName": "ES_AUTO_BACKUP",
"InstanceId": "es-f1m2enyd",
"Snapshots": [
{
"SnapshotName": "快照名称1",
"Uuid": "OPoARqBBSKuEIdLPdp7_dA",
"Version": "7.14.2",
"Indices": [
"honorboardcloud-sit-logs",
"pushtoken_meta_info_v1_auto",
".ds-logs-generic-default-2023.10.11-000002"
],
"State": "SUCCESS",
"StartTime": "2023-10-30T18:00:45.746Z",
"EndTime": "2023-10-30T18:14:46.707Z",
"DurationInMillis": 840961,
"TotalShards": 316,
"FailedShards": 0,
"SuccessfulShards": 316,
"Failures": [
{
"Index": "快照备份失败的索引名称",
"ShardId": 0,
"Reason": "快照失败的原因"
},
{
"Index": "快照备份失败的索引名称",
"ShardId": 0,
"Reason": "快照失败的原因"
}
]
},
{
"SnapshotName": "快照名称1",
"Uuid": "OPoARqBBSKuEIdLPdp7_dA",
"Version": "7.14.2",
"Indices": [
"honorboardcloud-sit-logs",
"pushtoken_meta_info_v1_auto",
".ds-logs-generic-default-2023.10.11-000002"
],
"State": "SUCCESS",
"StartTime": "2023-10-30T18:00:45.746Z",
"EndTime": "2023-10-30T18:14:46.707Z",
```

```
"DurationInMillis": 840961,
"TotalShards": 316,
"FailedShards": 0,
"SuccessfulShards": 316,
"Failures": [
{
"Index": "快照备份失败的索引名称",
"ShardId": 0,
"Reason": "快照失败的原因"
},
{
"Index": "快照备份失败的索引名称",
"ShardId": 0,
"Reason": "快照失败的原因"
}
]
}
]
}
}
```

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.CosInfoNotFound	COS相关信息获取失败。

删除快照仓库里备份的快照

最近更新时间：2025-04-25 01:31:45

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

删除快照仓库里备份的快照

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：DeleteClusterSnapshot。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	集群实例Id，格式：es-xxxx 示例值：es-xxxx
RepositoryName	是	String	快照仓库名称 示例值：BACKUP
SnapshotName	是	String	集群快照名称 示例值：test_snapshot

3. 输出参数

参数名称	类型	描述
InstanceId	String	集群id 示例值：es-xxxx
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 DeleteClusterSnapshot 删除集群快照数据

输入示例

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

{
"InstanceId": "es-al4radfu",
"RepositoryName": "ES_AUTO_BACKUP",
```

```
"SnapshotName": "es-al4radfu_20231120"
}
```

输出示例

```
{
"Response": {
"InstanceId": "es-al4radfu",
"RequestId": "xxx-xxxx"
}
}
```

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。

创建集群快照

最近更新时间：2025-04-25 01:31:46

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

集群快照手动创建

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：CreateClusterSnapshot。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例名称 示例值：es-xxx
SnapshotName	是	String	快照名称 示例值：test-snapshot_20220202
Indices	否	String	索引名称 示例值：index1,index2

3. 输出参数

参数名称	类型	描述
InstanceId	String	实例名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：es-xxxx
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 demo

输入示例

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

{
"InstanceId": "es-xxx",
"SnapshotName": "es-xxx-test_20240912",
```

```
"Indices": "kibana_sample_data_flights"
}
```

输出示例

```
{
"Response": {
"InstanceId": "es-xxx",
"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
FailedOperation.ErrorClusterState	集群状态错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidUin	Uin的取值和预期不符。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。

Serverless实例相关接口
创建Serverless索引项目空间(new)

最近更新时间：2025-06-26 01:20:39

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

创建Serverless索引空间

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：CreateServerlessSpaceV2。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
VpcInfo.N	是	Array of VpcInfo	vpc信息
SpaceName	否	String	索引空间名 示例值：space-feafnij
Zone	否	String	空间名称 示例值：ap-guangzhou-1
KibanaWhiteIpList.N	否	Array of String	白名单列表 示例值：["127.0.0.1"]
ZoneId	否	Integer	空间id 示例值：100001
TagList.N	否	Array of TagInfo	标签信息

3. 输出参数

参数名称	类型	描述
SpaceId	String	空间ID 示例值：space-xxxxxxxx
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 示例

输入示例

```
POST / HTTP/1.1
Host: es.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: CreateServerlessSpaceV2
<公共请求参数>

{
"SpaceName": "spaceName",
"VpcInfo": [
{
"VpcId": "vpc-feafeaf",
"SubnetId": "subnet-feafaffg",
"VpcUid": 1,
"SubnetUid": 1,
"AvailableIpAddressCount": 0
}
],
"Zone": "ap-guangzhou-1",
"KibanaWhiteIpList": [
"127.0.0.1"
],
"ZoneId": 100001
}
```

输出示例

```
{
"Response": {
"SpaceId": "space-feafaefgg",
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation	操作失败。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.SpaceNameExist	空间名已存在。
InvalidParameter.SpaceNameInvalid	空间名不合法。
LimitExceeded	超过配额限制。
LimitExceeded.IndexCount	空间索引数超过配额。
LimitExceeded.IndexCreate	空间索引每日创建数超过配额。
LimitExceeded.SpaceCount	空间数超过配额。
LimitExceeded.SpaceCreate	空间每日创建数超过配额。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.Clb	clb资源不足
ResourceInsufficient.Es	es资源不足
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。

版权所有：腾讯云计算（北京）有限责任公司

删除Serverless索引

最近更新时间：2025-06-26 01:20:39

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

删除Serverless索引

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：DeleteServerlessInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
InstanceId	是	String	serverless实例ID 示例值：index-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 删除Serverless实例

删除Serverless实例

输入示例

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

{
"InstanceId": "index-fwafefaef"
}
```

输出示例

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

```
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidServerlessId	ES serverless的serverlessId未找到
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。

版权所有：腾讯云计算（北京）有限责任公司

更新Serverless索引空间

最近更新时间：2025-06-26 01:20:38

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新Serverless索引空间

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：UpdateServerlessSpace。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceId	是	String	Serveless索引空间ID 示例值：space-abcdefgh
SpaceName	否	String	Serveless索引空间名 示例值：spaceName
KibanaPrivateAccess	否	String	kibana内网开关 示例值：OPEN
KibanaPublicAccess	否	String	kibana公网开关 示例值：OPEN
KibanaPublicAcl	否	EsAcl	kibana公网白名单

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 更新Serverless索引空间名

输入示例

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

{
```

```
"SpaceName": "242faff",
"SpaceId": "space-xxxxxx"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation	操作失败。
FailedOperation.AccessControlList	更新ACL失败
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.ErrorClusterStateUnhealth	集群状态不健康。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidNodeNames	节点名称列表NodeNames的取值和预期不符。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。

版权所有：腾讯云计算（北京）有限责任公司

错误码	描述
InvalidParameter.InvalidPublicAccess	外网访问PublicAccess的取值和预期不符。
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidSpaceId	ES Serverless的spaceId未找到
InvalidParameter.InvalidSpaceNames	space的名称错误
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient.SubnetIp	可用的子网IP不足。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。

版权所有：腾讯云计算（北京）有限责任公司

创建Serverless索引

最近更新时间：2025-06-26 01:20:40

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

创建Serverless索引

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：CreateServerlessInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
IndexName	是	String	索引名，需以-AppId结尾 示例值：test-12345678
Zone	否	String	可用区 示例值：ap-guangzhou-3
VpcId	否	String	私有网络ID 示例值：vpc-xxxxxx
SubnetId	否	String	子网ID 示例值：subnet-xxxxxx
IndexMetaJson	否	String	创建的索引元数据JSON，如mappings、settings 示例值："{"mappings":{"properties":{}},"options":{"expire.max_age":"3d","timestamp_field":"@timestamp"},"settings":{}}"
SpaceId	否	String	创建索引的空间ID 示例值：space-xxxxxxxx
Username	否	String	创建索引的用户名 示例值：sam
Password	否	String	创建索引的密码 示例值：fhauefFJNB
ServerlessDi	否	ServerlessDi	创建数据接入
AutoGetIp	否	Integer	是否自行添加白名单ip 示例值：0
TagList.N	否	Array of TagInfo	标签信息
KibanaWhiteIpList.N	否	Array of String	kibana公网白名单 示例值：["127.0.0.1"]

3. 输出参数

参数名称	类型	描述
InstanceId	String	实例ID 示例值：index-xxxxxxxx
DealName	String	订单号 注意：此字段可能返回 null，表示取不到有效值。 示例值：20250320235004448150491
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serverless实例

创建serverless实例

输入示例

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

{
"Zone": "ap-nanjing-1",
"VpcId": "vpc-dafafefe",
"SubnetId": "subnet-faefeffe",
"IndexName": "index_search",
"IndexMetaJson": "{\"mappings\":{\"properties\":{}},\"options\":{\"expire.max_age\":\"3d\",\"timestamp_field\":\"@timestamp\"},\"settings\":{}}",
"SpaceId": "space-dadaefef",
"Username": "user",
"Password": "affwaiUINbc"
}
```

输出示例

```
{
"Response": {
"InstanceId": "index-faefaf",
"DealName": "20250320244403348150491",
"RequestId": "6b044118-0566-11f0-b68a-5254001dee24 "
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.IndexNameExist	索引名已存在。
InvalidParameter.IndexNameInvalid	索引名不合法。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。
InvalidParameter.UsernameExist	用户名已存在。
LimitExceeded	超过配额限制。
LimitExceeded.IndexCount	空间索引数超过配额。
LimitExceeded.IndexCreate	空间索引每日创建数超过配额。
LimitExceeded.SpaceCount	空间数超过配额。
LimitExceeded.SpaceCreate	空间每日创建数超过配额。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。

查看Serverless空间子用户

最近更新时间：2025-06-26 01:20:39

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查看Serverless空间子用户

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：DescribeServerlessSpaceUser。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceId	是	String	空间的ID 示例值：space-xxxxxxxx
Offset	否	Integer	游标 示例值：1
Limit	否	Integer	页条数 示例值：10
UserNames.N	否	Array of String	用户名列表过滤 示例值：admin
UserTypes.N	否	Array of Integer	用户类型 示例值：0
PrivilegeTypes.N	否	Array of Integer	权限类型 示例值：0

3. 输出参数

参数名称	类型	描述
ServerlessSpaceUsers	Array of ServerlessSpaceUser	用户数组
TotalCount	Integer	用户总数 示例值：10
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 示例

输入示例

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

{
"SpaceId": "space-feafaef",
"Offset": 1,
"Limit": 10,
"UserNames": [
"admin"
],
"UserTypes": [
0
],
"PrivilegeTypes": [
0
]
}
```

输出示例

```
{
"Response": {
"ServerlessSpaceUsers": [
{
"Username": "admin",
"Password": "fwaifjiajf",
"CreateTime": "2023-06-22 00:52:58",
"Status": 1,
"IndicesScope": [
"indexName"
],
"PrivilegeType": 0
}
],
"TotalCount": 1,
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.GetUser	获取user失败
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidSpaceId	ES Serverless的spaceId未找到
InvalidParameter.UsernameExist	用户名已存在。
LimitExceeded	超过配额限制。
LimitExceeded.IndexCount	空间索引数超过配额。
LimitExceeded.IndexCreate	空间索引每日创建数超过配额。
LimitExceeded.SpaceCount	空间数超过配额。
LimitExceeded.SpaceCreate	空间每日创建数超过配额。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。

版权所有：腾讯云计算（北京）有限责任公司

删除Serverless空间子用户

最近更新时间：2025-04-25 01:31:47

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

删除Serverless空间子用户

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：DeleteServerlessSpaceUser。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceId	是	String	空间的ID 示例值：space-xxxxxxxx
Username	是	String	创建索引的用户名 示例值：admin

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 示例

输入示例

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

{
"SpaceId": "space-efafeafe",
"Username": "admin"
}
```

输出示例

```
{
"Response": {
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NotAuthenticated	用户未实名认证。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.UsernameExist	用户名已存在。
LimitExceeded	超过配额限制。
LimitExceeded.IndexCount	空间索引数超过配额。
LimitExceeded.IndexCreate	空间索引每日创建数超过配额。
LimitExceeded.SpaceCount	空间数超过配额。
LimitExceeded.SpaceCreate	空间每日创建数超过配额。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。

Serverless获取索引列表

最近更新时间：2025-06-26 01:20:39

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

Serverless获取索引列表

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：DescribeServerlessInstances。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
InstanceIds.N	否	Array of String	索引集群ID 示例值：["index-abcdefgh"]
IndexNames.N	否	Array of String	索引名 示例值：["test"]
Offset	否	Integer	分页起始位置 示例值：10
Limit	否	Integer	一页展示数量 示例值：20
OrderBy	否	String	排序字段，支持索引名：IndexName、索引存储量：IndexStorage、索引创建时间：IndexCreateTime 示例值：storage
IndexStatusList.N	否	Array of String	过滤索引状态 示例值：["GREEN"]
Order	否	String	排序顺序，支持asc、desc，默认为desc 示例值：asc
SpaceIds.N	否	Array of String	索引空间ID列表 示例值：[space-12345]
DiSourceTypes.N	否	Array of String	数据链路数据源类型 示例值：["cvm_collector","tke_collector"]
TagList.N	否	Array of TagInfo	标签信息

3. 输出参数

参数名称	类型	描述
IndexMetaFields	Array of ServerlessIndexMetaField	索引元数据字段 注意：此字段可能返回 null，表示取不到有效值。

参数名称	类型	描述
TotalCount	Integer	查询总数 注意：此字段可能返回 null，表示取不到有效值。 示例值：10
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 Serverless获取索引列表

Serverless获取索引列表

输入示例

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

{
"IndexNames": [
"indexName"
],
"InstanceIds": [
"index-abcdefgh"
],
"Limit": 0,
"Offset": 10
}
```

输出示例

```
{
"Response": {
"IndexMetaFields": [
{
"AppId": 1257780094,
"InstanceId": "index-2v3kt0yr",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "curl-datalink-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:34:55",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
```

```
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-2v3kt0yr.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-2v3kt0yr.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-2s8pkoa7",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "beats-tke-datalink-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:34:20",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-2s8pkoa7.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-2s8pkoa7.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-7th48x7l",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "beats-cvm-datalink-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:33:48",
"IndexDocs": 0,
```

```
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-7th48x7l.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-7th48x7l.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-brjddwvb",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "tke-data-link-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:30:26",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-brjddwvb.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-brjddwvb.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
```

```
"AppId": 1257780094,
"InstanceId": "index-k75a66tp",
"SpaceId": "space-0gyg87rx",
"SpaceName": "price-test",
"IndexName": "ychenjiang-test2-0gyg87rx",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:29:47",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-1d5dkwwx",
"SubnetUid": "subnet-267c2qvq",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-k75a66tp.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-k75a66tp.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-o8otig09",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "cvm-datalink-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 14:23:02",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
```

```
"Password": ""
},
"KibanaUrl": "https://index-o8otig09.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-o8otig09.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-gh5rrdhr",
"SpaceId": "space-0gyg87rx",
"SpaceName": "price-test",
"IndexName": "faegagag-0gyg87rx",
"IndexMetaJson": "",
"IndexStorage": 0,
"IndexCreateTime": "2023-11-23 12:04:48",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-gh5rrdhr.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-gh5rrdhr.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-65q7dsff",
"SpaceId": "space-0gyg87rx",
"SpaceName": "price-test",
"IndexName": "haoa-test3334-0gyg87rx",
"IndexMetaJson": "",
"IndexStorage": 6422689,
"IndexCreateTime": "2023-11-23 11:53:04",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
```

```
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-1d5dkwwx",
"SubnetUid": "subnet-267c2qvq",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-65q7dsff.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-65q7dsff.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-irah4s23",
"SpaceId": "space-mdo5a8bf",
"SpaceName": "tom-test",
"IndexName": "cvm-source-mdo5a8bf",
"IndexMetaJson": "",
"IndexStorage": 204964,
"IndexCreateTime": "2023-11-23 11:39:28",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-444yjczv",
"SubnetUid": "subnet-mrezecu2",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-irah4s23.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-irah4s23.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
},
{
"AppId": 1257780094,
"InstanceId": "index-8s7iinrd",
"SpaceId": "space-0gyg87rx",
"SpaceName": "price-test",
"IndexName": "haoa-test44-0gyg87rx",
"IndexMetaJson": "",
```

```
"IndexStorage": 6072662,
"IndexCreateTime": "2023-11-23 11:15:47",
"IndexDocs": 0,
"Status": 1,
"IndexSettingsField": {
"NumberOfShards": "",
"RefreshInterval": "30s"
},
"IndexOptionsField": {
"ExpireMaxAge": "30d",
"TimestampField": "@timestamp"
},
"IndexNetworkField": {
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-7",
"VpcUid": "vpc-ld5dkwwx",
"SubnetUid": "subnet-267c2gvq",
"Username": "",
"Password": ""
},
"KibanaUrl": "https://index-8s7iinrd.kibana.myserverlessindex.com:5601",
"KibanaPrivateUrl": "",
"IndexAccessUrl": "index-8s7iinrd.ap-guangzhou.myserverlessindex.com",
"StorageType": 0,
"TagList": []
}
],
"TotalCount": 20,
"RequestId": "a75f987c-5f03-4fec-bbb8-be132b57666b"
}
}
```


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.UnAuthorizedOperation	操作未授权。
FailedOperation	操作失败。
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.ErrorClusterStateUnhealth	集群状态不健康。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidServerlessId	ES serverless的serverlessId未找到
InvalidParameter.InvalidSpaceId	ES Serverless的spaceId未找到
InvalidParameter.InvalidTagInfo	节点标签信息TagInfo的取值和预期不符。
InvalidParameter.InvalidTagList	节点标签信息列表TagList的取值和预期不符。
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceNotFound	资源不存在。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取Serverless索引空间列表

最近更新时间：2025-06-26 01:20:38

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取Serverless索引空间列表

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：DescribeServerlessSpaces。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceIds.N	否	Array of String	过滤的空间ID 示例值：[space-xxxxxxx]
SpaceNames.N	否	Array of String	过滤的空间名 示例值：[test]
Order	否	String	排序顺序，支持升序asc、降序desc 示例值：desc
OrderBy	否	String	排序字段，支持空间创建时间SpaceCreateTime 示例值：SpaceCreateTime
VpcIds.N	否	Array of String	vpcId信息数组 示例值：["vpc-abfeaefm"]
Offset	否	Integer	分页起始 示例值：1
Limit	否	Integer	分页条数 示例值：10
TagList.N	否	Array of TagInfo	标签信息

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	查询总数 注意：此字段可能返回 null，表示取不到有效值。 示例值：10
ServerlessSpaces	Array of ServerlessSpace	Serverless空间信息列表 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 获取Serverless索引空间列表

获取Serverless索引空间列表

输入示例

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

{}
```

输出示例

```
{
"Response": {
"TotalCount": 0,
"ServerlessSpaces": [
{
"SpaceId": "space-eafhiefi",
"SpaceName": "spaceName",
"Status": 1,
"CreateTime": "2024-12-11 17:50:05",
"IndexCount": 1
}
],
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation	操作失败。
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.ErrorClusterStateUnhealth	集群状态不健康。
FailedOperation.GetTagInfoError	获取实例的标签列表错误。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

版权所有：腾讯云计算（北京）有限责任公司

更新Serverless索引

最近更新时间：2025-06-26 01:20:38

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

更新Serverless索引

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：UpdateServerlessInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
InstanceId	是	String	Serveless实例ID 示例值：index-abcdefgh
UpdateMetaJson	否	String	更新的索引元数据JSON，如mappings、settings 示例值："{"mappings":{"properties":{}},"options":{"expire.max_age":"7d"},"settings":{}}"
Password	否	String	实例访问密码 示例值：fwjaifAFJElfj
KibanaPrivateAccess	否	String	开启kibana内网访问，如OPEN、CLOSE 示例值：OPEN
KibanaPublicAccess	否	String	开启kibana外网访问，如OPEN、CLOSE 示例值：OPEN
KibanaPublicAcl	否	KibanaPublicAcl	访问控制列表

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 更新索引元数据

更新索引元数据---- 添加ResourceInUse.IsolatedStatus错误码

输入示例

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

```
<公共请求参数>

{
"UpdateMetaJson": "{\"mappings\":{\"properties\":{}},\"options\":{\"expire.max_age\":\"7d\"},\"settings\":{}}",
"InstanceId": "index-asdasd"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.UnAuthorizedOperation	操作未授权。
FailedOperation	操作失败。
FailedOperation.AccessControlList	更新ACL失败
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.ErrorClusterStateUnhealth	集群状态不健康。
FailedOperation.SpaceLanguage	变更space的kibana语言信息失败
FailedOperation.UnsupportedOpenLowFrequency	不支持开启低频存储

错误码	描述
FailedOperation.UpdateVpc	变更space的vpc信息失败
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidEsACL	EsAcl的取值和预期不符。
InvalidParameter.InvalidIndexMetaJson	创建的索引元数据JSON IndexMetaJson的取值和预期不符。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。
InvalidParameter.InvalidPrivateAccess	内网访问PrivateAccess的取值和预期不符。
InvalidParameter.InvalidPublicAccess	外网访问PublicAccess的取值和预期不符。
InvalidParameter.InvalidServerlessId	ES serverless的serverlessId未找到
InvalidParameter.InvalidUpdateMetaJson	更新的索引元数据json UpdateMetaJson的取值和预期不符。
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInUse.IsolatedStatus	索引状态处于隔离状态，不能进行其他操作。
UnauthorizedOperation	未授权操作。
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

版权所有：腾讯云计算（北京）有限责任公司

获取实例对应的监控指标

最近更新时间：2025-04-25 01:31:47

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取serverless实例对应指标，获取space维度时不需要传入indexid，获取index时不需要传入spaceid
获取一段时间时间范围内的指标数据

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：DescribeServerlessMetrics。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceId	否	String	space空间id 示例值：space-xxxxxx
IndexId	否	String	index索引id 示例值：index-ex2ap8v0
MetricType.N	否	Array of String	指标类型，暂时只支持Storage(存储大小),AllMetric(所有存储指标：索引流量、存储大小、文档数量、读请求和写请求) 示例值：["Storage"]
DurationType	否	Integer	时间长度类型DurationType(1: 3小时, 2: 昨天1天,3: 今日0点到现在) 示例值：1
BatchIndexList.N	否	Array of String	索引数据 示例值：["index-4y85drgw"]

3. 输出参数

参数名称	类型	描述
Storage	Float	storage指标值，单位byte 示例值：2262158.88
IndexTraffic	Float	IndexTraffic指标值，单位byte 示例值：2262158.88
ReadReqTimes	Integer	读请求数，单位次数 示例值：100
WriteReqTimes	Integer	写请求数，单位次数 示例值：100
DocCount	Integer	文档数量，单位个数 示例值：100
MetricMapList	Array of	指标数据数据

参数名称	类型	描述
	MetricMapByIndexId	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 获取serverless space维度过去5min钟新增存储指标值

输入示例

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

{
"SpaceId": "space-18jf9xxx",
"IndexId": "index-xxxxxx",
"MetricType": [
"Storage"
]
}
```

输出示例

```
{
"Response": {
"RequestId": "30204658-3518-11ef-979c-52540xxxxx",
"Storage": 2262158.88
}
}
```

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。

space维度的kibana获取登录token

最近更新时间：2025-06-26 01:20:38

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

space维度的kibana获取登录token

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：DescribeSpaceKibanaTools。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
SpaceId	是	String	space的ID 示例值：space-xxx

3. 输出参数

参数名称	类型	描述
KibanaToken	String	该token用于登录内嵌kibana 示例值：eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.
ExpireTime	Integer	token的过期时间 示例值：1742806292
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 kibana内嵌登录token获取

用于登录space维度的内嵌的kibana

输入示例

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

{
"SpaceId": "space-eafijiaef"
}
```

输出示例

```
{
"Response": {
"KibanaToken": "feijfioajefbifiejiaUFHieuf",
"ExpireTime": 1742806292,
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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	操作失败。

查询快照信息接口

最近更新时间：2025-06-26 01:20:38

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询快照信息接口

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：DescribeUserCosSnapshotList。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
CosBucket	否	String	cos桶名 示例值：user-cos
BasePath	否	String	bucket 桶下的备份路径 示例值：dir/
ClusterInstanceId	否	String	云上集群迁移集群名 示例值：es-feaifhie

3. 输出参数

参数名称	类型	描述
CosSnapshotInfoList	Array of CosSnapShotInfo	cos 快照信息列表 示例值：[{"CosBucket":"es-ap-beijing"}]
TotalCount	Integer	cos 快照数量 示例值：1
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 查询快照信息接口

输入示例

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

```
{
"CosBucket": "user-cos",
"BasePath": "dir/",
"ClusterInstanceId": "es-feaffea"
}
```

输出示例

```
{
"Response": {
"CosSnapshotInfoList": [
{
"CosBucket": "user-cos",
"BasePath": "dir/",
"SnapshotName": "testSnapshot",
"State": "SUCCESS",
"Version": "7.14.2",
"CommonIndexArr": [
{
"IndexName": "testindex",
"IsShardComplete": 0
}
],
"DataStreamArr": [
{
"DataStreamName": "index",
"IsShardComplete": 0
}
]
}
],
"TotalCount": 10,
"RequestId": "iuhifaeui-feaihfhfa-feaijni-fieai"
}
}
```

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
FailedOperation.GetUserCosSnapshotList	获取快照列表失败
InvalidParameter	参数错误。

cos迁移流程

最近更新时间：2025-06-26 01:20:40

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

cos迁移流程

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：CreateCosMigrateToServerlessInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
Snapshot	是	String	快照名 示例值：test_snapshot
ServerlessId	是	String	索引 id 示例值：index-fwafafaf
CosBucket	否	String	cos 桶名 示例值：user-cos
BasePath	否	String	BasePath 路径 示例值：dir/
ClusterInstanceId	否	String	云上集群 id 示例值：es-feafaefef
CommonIndexArr.N	否	Array of String	待迁移普通索引名列表 示例值：["test1"]
DataStreamArr.N	否	Array of String	待迁移自治索引名列表 示例值：["test1"]

3. 输出参数

参数名称	类型	描述
TaskId	String	迁移 taskid 示例值：1
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 cos迁移流程

输入示例

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

{
"CosBucket": "user-cos",
"BasePath": "dir/",
"Snapshot": "test_snapshot",
"ServerlessId": "index-faafeeaf",
"ClusterInstanceId": "es-feafaefe",
"CommonIndexArr": [
"test1"
],
"DataStreamArr": [
"test1"
]
}
```

输出示例

```
{
"Response": {
"TaskId": "cosMigrateTask-81zl9i82",
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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	操作失败。
FailedOperation.CosMigrate	cos融合迁移失败
InvalidParameter.InvalidServerlessId	ES serverless的serverlessId未找到
ResourceNotFound.EsInfoNotFound	集群信息获取失败
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。

检查cos迁移索引元数据

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

检查cos迁移索引元数据

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：CheckMigrateIndexMetaData。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表，本接口仅支持其中的: ap-beijing, ap-guangzhou, ap-hongkong, ap-nanjing, ap-shanghai, ap-singapore, ap-tokyo, na-ashburn 。
ServerlessId	是	String	索引 id 示例值：index-faefaefg
Snapshot	是	String	快照名 示例值：snapshot
CosBucket	否	String	Cos桶名 示例值：user-cos-name
BasePath	否	String	BasePath路径 示例值：dir/
ClusterInstanceId	否	String	云上集群名 示例值：es-fwajfwii
CommonIndexArr.N	否	Array of String	普通索引名列表 示例值：["index1", "index2"]
DataStreamArr.N	否	Array of String	自治索引名列表 示例值：["index1", "index2"]

3. 输出参数

参数名称	类型	描述
MappingTimeFieldCheckFailedIndexArr	Array of String	不存在于目标索引时间字段相同的字段 示例值：[""]
MappingTimeTypeCheckFailedIndexArr	Array of String	@timestamp不为date类型，与目标索引时间字段冲突 示例值：[""]
SettingCheckFailedIndexArr	Array of String	索引的创建时间不在 serverless的存储周期内 示例值：[""]
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 检查cos迁移索引元数据

输入示例

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

{
"CosBucket": "user-cos",
"BasePath": "dir/",
"ServerlessId": "index-feafagg",
"Snapshot": "snapshot",
"ClusterInstanceId": "es-feafegg",
"CommonIndexArr": [
"index"
],
"DataStreamArr": [
"index"
]
}
```

输出示例

```
{
"Response": {
"MappingTimeFieldCheckFailedIndexArr": [
"index"
],
"MappingTimeTypeCheckFailedIndexArr": [
"index"
],
"SettingCheckFailedIndexArr": [
"index"
],
"RequestId": "40f4f780-9969-42f9-8bd9-ccf0d1f2591a"
}
}
```

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

版权所有：腾讯云计算（北京）有限责任公司

- 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。

Logstash相关接口
创建Logstash实例

最近更新时间：2025-05-21 01:18:57

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于创建Logstash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：CreateLogstashInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceName	是	String	实例名称（1-50 个英文、汉字、数字、连接线-或下划线_） 示例值：logstash_test
Zone	是	String	可用区 示例值：ap-guangzhou-3
LogstashVersion	是	String	实例版本（支持"6.8.13"、"7.10.1"） 示例值：6.8.13
VpcId	是	String	私有网络ID 示例值：vpc-xxxxxx
SubnetId	是	String	子网ID 示例值：subnet-xxxxxx
NodeNum	否	Integer	节点数量（2-50个） 示例值：3
ChargeType	否	String	计费类型 • PREPAID：预付费，即包年包月 • POSTPAID_BY_HOUR：按小时后付费 默认值POSTPAID_BY_HOUR 示例值：POSTPAID_BY_HOUR
ChargePeriod	否	Integer	包年包月购买时长（单位由参数TimeUnit决定） 示例值：1
TimeUnit	否	String	计费时长单位（ChargeType为PREPAID时需要设置，默认值为“m”，表示月，当前只支持“m”） 示例值：m
AutoVoucher	否	Integer	是否自动使用代金券 • 0：不自动使用 • 1：自动使用 默认值0 示例值：1

参数名称	必选	类型	描述
VoucherIds.N	否	Array of String	代金券ID列表（目前仅支持指定一张代金券） 示例值：[""]
RenewFlag	否	String	自动续费标识 • RENEW_FLAG_AUTO：自动续费 • RENEW_FLAG_MANUAL：不自动续费，用户手动续费 ChargeType为PREPAID时需要设置，如不传递该参数，普通用户默认不自动续费，SVIP用户自动续费 示例值：RENEW_FLAG_DEFAULT
NodeType	否	String	节点规格 • LOGSTASH.S1.SMALL2：1核2G • LOGSTASH.S1.MEDIUM4：2核4G • LOGSTASH.S1.MEDIUM8：2核8G • LOGSTASH.S1.LARGE16：4核16G • LOGSTASH.S1.2XLARGE32：8核32G • LOGSTASH.S1.4XLARGE32：16核32G • LOGSTASH.S1.4XLARGE64：16核64G 示例值：LOGSTASH.S1.SMALL2
DiskType	否	String	节点磁盘类型 • CLOUD_SSD：SSD云硬盘 • CLOUD_PREMIUM：高硬能云硬盘 默认值CLOUD_SSD 示例值：CLOUD_SSD
DiskSize	否	Integer	节点磁盘容量（单位GB） 示例值：100
LicenseType	否	String	License类型 • oss：开源版 • xpack：xpack版 默认值xpack 示例值：xpack
TagList.N	否	Array of TagInfo	标签信息列表
OperationDuration	否	OperationDuration	可维护时间段
MultiZoneInfo.N	否	Array of ZoneDetail	多可用区部署时可用区的详细信息
DeployMode	否	Integer	部署模式，0：单可用区、1：多可用区 示例值：0

3. 输出参数

参数名称	类型	描述
InstanceId	String	实例ID 示例值：ls-xxxxxx
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Logstash实例

根据输入参数创建Logstash实例

输入示例

```
POST / HTTP/1.1
Host: es.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: CreateLogstashInstance
<公共请求参数>

{
"VpcId": "vpc-xxxxxx",
"LogstashVersion": "6.8.13",
"Zone": "ap-guangzhou-3",
"DiskType": "CLOUD_SSD",
"NodeNum": "3",
"DiskSize": "100",
"ChargeType": "POSTPAID_BY_HOUR",
"SubnetId": "subnet-xxxxxx",
"NodeType": "LOGSTASH.S1.SMALL2",
"InstanceName": "logstash_test"
}
```

输出示例

```
{
"Response": {
"InstanceId": "ls-xxxxxx",
"RequestId": "d7b76d5e-ad7d-4abd-b3b2-43b96dxxxxxx"
}
}
```

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.UnAuthCreateInstance	当前用户对创建集群实例操作未授权，请添加CAM权限

错误码	描述
AuthFailure.UnAuthDescribeInstances	当前用户对查询集群实例操作未授权，请添加CAM权限。
AuthFailure.UnAuthorizedOperation	操作未授权。
FailedOperation.CloseDataTier	关闭数据层失败，至少要保留热层和温层其中一层。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.EsDictionaryInfoError	由于EsDictionaryInfo错误，不允许操纵。
FailedOperation.FileNameError	文件名不符合规范。
FailedOperation.FileSizeError	文件大小不符合要求。
FailedOperation.GetTagInfoError	获取实例的标签列表错误。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NodeNumAndZoneError	节点数和可用区的信息不符合对应操作的规则。
FailedOperation.Password	解码Password时出错。
FailedOperation.RefundError	该实例不满足退费条件。
FailedOperation.RequestTimeOut	请求超时。
FailedOperation.TradeSignError	生成询价或者下单签名有误。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.GC	Gc类型的取值和预期不符。
InvalidParameter.InvalidAutoVoucher	是否自动使用代金券AutoVoucher的取值和预期不符。
InvalidParameter.InvalidBackends	Backends的取值和预期不符。
InvalidParameter.InvalidDeployMode	集群部署方式InvalidDeployMode的取值和预期不符。
InvalidParameter.InvalidDiskEnhance	是否启用增强型ssd云盘DiskEnhance的取值和预期不符。
InvalidParameter.InvalidDns	内网域名解析服务器IP Dns的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexMetaJson	创建的索引元数据JSON IndexMetaJson的取值和预期不符。
InvalidParameter.InvalidIndexName	索引名IndexName的取值和预期不符。
InvalidParameter.InvalidIndexOptionsField	索引自治字段IndexOptionsField的取值和预期不符。
InvalidParameter.InvalidIndexPolicyField	索引生命周期字段IndexPolicyField的取值和预期不符。
InvalidParameter.InvalidIndexSettingsField	索引配置字段IndexSettingsField的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceOperationDurations	集群可维护时间段和集群ID InstanceOperationDurations的取值和预期不符。
InvalidParameter.InvalidLogType	日志类型LogType的取值和预期不符。
InvalidParameter.InvalidLogstashVersion	实例版本LogstashVersion的取值和预期不符。
InvalidParameter.InvalidNodeNames	节点名称列表NodeNames的取值和预期不符。
InvalidParameter.InvalidNodeTypeStorageInfo	温热层数据结构NodeTypeInfos的取值和预期不符。
InvalidParameter.InvalidOperationDuration	可维护时间段OperationDuration的取值和预期不符。
InvalidParameter.InvalidPara	需要签名的参数Para的取值和预期不符。
InvalidParameter.InvalidRestartMode	可选重启模式RestartMode的取值和预期不符。

错误码	描述
InvalidParameter.InvalidRestartType	重启方式RestartType的取值和预期不符。
InvalidParameter.InvalidSampleJson	数据样例JSON SampleJson的取值和预期不符。
InvalidParameter.InvalidSecurityGroupIds	安全组id列表SecurityGroupIds的取值和预期不符。
InvalidParameter.InvalidSubnetUidList	虚拟子网络统一ID列表的取值和预期不符。
InvalidParameter.InvalidTargetIndexName	目标索引名TargetIndexName的取值和预期不符。
InvalidParameter.InvalidTargetNodeTypes	接收请求的目标节点类型列表InvalidTargetNodeTypes的取值和预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
InvalidParameter.InvalidTraceUuid	调用链Id TraceUuid的取值和预期不符。
InvalidParameter.InvalidType	节点类型Type的取值和预期不符。
InvalidParameter.InvalidUpdateMetaJson	更新的索引元数据json UpdateMetaJson的取值和预期不符。
InvalidParameter.InvalidVoucherIds	代金券ID列表VoucherIds的取值和预期不符。
InvalidParameterValue.ChargePeriod	包年包月购买时长ChargePeriod的取值有误。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
InvalidParameterValue.CheckOnly	是否只做升级检查CheckOnly的取值有误。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.EnableDoubleEni	是否允许双网卡EnableDoubleEni的值有误。
InvalidParameterValue.EsPort	端口EsPort的取值有误。
InvalidParameterValue.EsVip	EsVip的取值有误。
InvalidParameterValue.InstallBundleList	要安装的bundle信息列表InstallBundleList的取值有误。
InvalidParameterValue.InstallPluginList	为集群稳定起见，当前只支持同时安装1个用户插件。
InvalidParameterValue.InvalidJDK	Jdk类型的取值和预期不符。
InvalidParameterValue.PluginType	插件类型PluginType的取值有误。
InvalidParameterValue.RemoveBundleList	要卸载的bundle名列表RemoveBundleList的取值有误。
InvalidParameterValue.RemovePluginList	为集群稳定起见，当前只支持同时卸载1个用户插件。
InvalidParameterValue.RenewFlag	自动续费标识RenewFlag的参数取值有误。
InvalidParameterValue.TargetPath	要安装的bundle的zip包解压所在的路径TargetPath的取值有误。
InvalidParameterValue.TimeUnit	计费时长单位TimeUnit的取值有误。
InvalidParameterValue.UpgradeMode	升级方式UpgradeMode的取值有误。
LimitExceeded.ClusterNum	该账号下的集群数超过限额。
LimitExceeded.DiagnoseCount	超过当天智能诊断最大次数。
LimitExceeded.DiskCount	节点磁盘块数超过最大限额。
LimitExceeded.NodeNumOrIndices	节点数量或索引存储量超过限额。
LimitExceeded.PluginInstall	超过最大自定义插件数量，请先删除部分自定义插件。
RegionError	地域错误
ResourceInUse	资源被占用。
ResourceInUse.Diagnose	存在诊断中的作业，请等待作业诊断完成后重试。
ResourceInUse.Order	订单被锁定。

错误码	描述
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.CVM	计算资源不足，可更换可用区或调整机型。
ResourceInsufficient.CVMQuota	cvm份额不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
ResourceNotFound.AccountInfoNotFound	账户信息未找到。
ResourceNotFound.BaradMetricNotFound	巴拉多（barad）指标资源未找到。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
ResourceNotFound.CLBNotFound	负载均衡信息未找到。
ResourceNotFound.CosInfoNotFound	COS相关信息获取失败。
ResourceNotFound.DiagnoseNotFound	智能诊断相关资源未找到。
ResourceNotFound.KMSNotFound	密钥管理系统资源获取失败。
ResourceNotFound.STSNotFound	获取安全凭证服务资源失败。
ResourceNotFound.SecurityGroupNotFound	安全组信息获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。
ResourceNotFound.VPCInfoNotFound	VPC资源获取失败。
UnsupportedOperation.AuditStatusConflict	审计日志的状态冲突。
UnsupportedOperation.CosBackUp	该版本不支持cos备份。集群需要重启来允许cos备份。
UnsupportedOperation.DiagnoseJob	不支持该智能诊断任务。
UnsupportedOperation.DiagnoseNotOpen	智能诊断未开启。
UnsupportedOperation.DiskUse	不支持的读硬盘配置。
UnsupportedOperation.EditListLength	不支持这个长度的editList的操作。
UnsupportedOperation.Migrate	不支持迁移操作。
UnsupportedOperation.PassLogstashId	用户同步日志的配置passLogstashId为空，不支持该操作。
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.Protocol	不支持对应Protocol的操作。
UnsupportedOperation.ReadRate	不支持的读取速度。
UnsupportedOperation.RestartMode	离线节点蓝绿变更存在风险，请直接选择重启“in-place”。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.UpdateDiskEncrypt	节点或磁盘数变更时不支持更新磁盘加密信息。
UnsupportedOperation.UpdateDiskType	不支持更新磁盘类型。
UnsupportedOperation.UpdateNodeNumAndNodeType	不支持同时更新节点个数（NodeNum）和节点规格（NodeType）。
UnsupportedOperation.WebServiceType	不支持的web组件。
UnsupportedOperation.WriteRate	不支持的写入速度。

删除Logstash实例

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于删除Logstash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：DeleteLogstashInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-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 删除Logstash实例

删除指定ID的Logstash实例

输入示例

```
https://es.tencentcloudapi.com/?Action=DeleteLogstashInstance
&InstanceId=ls-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。
UnsupportedOperation	操作不支持。

删除Logstash管道

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于批量删除Logstash管道

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：DeleteLogstashPipelines。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
PipelineIds.N	否	Array of String	管道ID列表 示例值：["pip_xxx"]

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Logstash管道

输入示例

```
https://es.tencentcloudapi.com/?Action=DeleteLogstashPipelines
&InstanceId=ls-xxxxxxxx
&PipelineIds.0=pip-xxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。

查询Logstash实例日志

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询用户该地域下符合条件的Logstash实例的日志

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：DescribeLogstashInstanceLogs。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
LogType	否	Integer	日志类型，默认值为1 • 1, 主日志 • 2, 慢日志 • 3, GC日志 示例值：1
SearchKey	否	String	搜索词，支持LUCENE语法，如 level:WARN、ip:1.1.1.1、message:test-index等 示例值：level:INFO OR level:WARN
StartTime	否	String	日志开始时间，格式为YYYY-MM-DD HH:MM:SS, 如2019-01-22 20:15:53 示例值：2019-01-22 20:15:53
EndTime	否	String	日志结束时间，格式为YYYY-MM-DD HH:MM:SS, 如2019-01-22 20:15:53 示例值：2019-01-22 21:15:53
Offset	否	Integer	分页起始值, 默认值为0 示例值：0
Limit	否	Integer	分页大小，默认值为100，最大值100 示例值：10
OrderByType	否	Integer	时间排序方式，默认值为0 • 0, 降序 • 1, 升序 示例值：1

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	返回的日志条数 示例值：10

参数名称	类型	描述
InstanceLogList	Array of InstanceLog	日志详细信息列表 示例值：[{}]
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 获取Logstash实例日志

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeLogstashInstanceLogs
&InstanceId=ls-f5mwm28u
&<公共请求参数>
```

输出示例


```
{
"Response": {
"TotalCount": 71633,
"InstanceLogList": [
{
"Time": "2019-01-22T10:45:36.220+08:00",
"Ip": "10.0.128.65",
"NodeID": "12123141441",
"Level": "INFO",
"Message": "[o.e.p.o.OPackActionFilter] [1547723102001286009] forbidden request: { ID:cdcxxxabc, TYP:MainRequest, USR:nul
l, BRS:false, ACT:cluster:monitor/main, OA:10.0.128.43, IDX:, MET:GET, PTH:/, CNT:<OMITTED, LENGTH=0>, HDR:content-length,
EFF:0 } Reason: null"
},
{
"Time": "2019-01-22T10:45:35.730+08:00",
"NodeID": "12123141442",
"Ip": "10.0.128.65",
"Level": "INFO",
"Message": "[o.e.p.o.OPackActionFilter] [1547723102001286009] forbidden request: { ID:1a8axxxabc, TYP:MainRequest, USR:nul
l, BRS:false, ACT:cluster:monitor/main, OA:10.0.128.73, IDX:, MET:GET, PTH:/, CNT:<OMITTED, LENGTH=0>, HDR:content-length,
EFF:0 } Reason: null"
}
],
"RequestId": "783d9290-dc60-4862-9340-10b632605374"
}
}
```


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.RequestTimeOut	请求超时。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameter.InvalidQueryString	无效的lucene语法
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。

获取Logstash实例列表

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询用户该地域下符合条件的所有Logstash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：DescribeLogstashInstances。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
Zone	否	String	实例所属可用区，不传则默认所有可用区 示例值：ap-guangzhou
InstanceIds.N	否	Array of String	实例ID列表 示例值：["ls-xxxxxxxx"]
InstanceNames.N	否	Array of String	实例名称列表 示例值：testnew
Offset	否	Integer	分页起始值, 默认值0 示例值：0
Limit	否	Integer	分页大小，默认值20 示例值：0
OrderByKey	否	Integer	排序字段 • 1：实例ID • 2：实例名称 • 3：可用区 • 4：创建时间 若orderKey未传递则按创建时间降序排序 示例值：1
OrderByType	否	Integer	排序方式 • 0：升序 • 1：降序 若传递了orderByKey未传递orderByType, 则默认升序 示例值：1
VpcIds.N	否	Array of String	VpcId 筛选项 示例值：vpc-imhajivf
TagList.N	否	Array of TagInfo	标签信息列表

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	返回的实例个数 示例值：2
InstanceList	Array of LogstashInstanceInfo	实例详细信息列表
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Logstash实例

根据给定条件查询符合条件的Logstash实例并返回详细信息

输入示例

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

{}
```

输出示例

```
{
"Response": {
"TotalCount": 2,
"InstanceList": [
{
"InstanceId": "ls-ee6o1285",
"InstanceName": "wuxxx-20200927-ls01",
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-3",
"AppId": 1257780094,
"Uin": "100007799884",
"VpcId": "vpc-ck9abamb",
"SubnetId": "subnet-gmmt588q",
"Status": 1,
"ChargeType": "POSTPAID_BY_HOUR",
"ChargePeriod": 0,
"RenewFlag": "",
"NodeType": "LOGSTASH.S1.MEDIUM4",
"NodeNum": 5,
"DiskType": "CLOUD_SSD",
"DiskSize": 100,
"LogstashVersion": "6.8.10",
"LicenseType": "xpack",
"YMLConfig": "",
"BindedESInstanceId": "",
"Nodes": [
{
"NodeId": "100003160102096307",
"Ip": "10.0.3.21",
"Port": 9600
```

```
},
{
"NodeId": "100003160102096308",
"Ip": "10.0.3.134",
"Port": 9600
},
{
"NodeId": "100003160102514909",
"Ip": "10.0.3.151",
"Port": 9600
},
{
"NodeId": "100003160102514910",
"Ip": "10.0.3.160",
"Port": 9600
},
{
"NodeId": "100003160102514911",
"Ip": "10.0.3.157",
"Port": 9600
}
],
"ExtendedFiles": [
{
"Name": "commons-codec-1.9.jar",
"Size": 1
}
],
"CpuNum": 1,
"MemSize": 1,
"OperationDuration": {
"TimeStart": "02:00",
"TimeZone": "UTC+8",
"Periods": [
1
],
"TimeEnd": "06:00"
},
"TagList": [
{
"TagKey": "产品",
"TagValue": "ES"
}
],
"CreateTime": "2020-09-25 16:02:43",
"UpdateTime": "2020-09-27 16:10:17",
"Deadline": "2017-12-04 00:00:00"
},
{
"InstanceId": "ls-g8k9pzj9",
"InstanceName": "wurong-20200915-ls01",
"Region": "ap-guangzhou",
"Zone": "ap-guangzhou-3",
"AppId": 1257780094,
"Uin": "100007799884",
"VpcId": "vpc-ck9abamb",
"SubnetId": "subnet-gmmt588q",
```

```
"Status": 1,
"ChargeType": "POSTPAID_BY_HOUR",
"ChargePeriod": 0,
"RenewFlag": "",
"NodeType": "LOGSTASH.S1.MEDIUM4",
"NodeNum": 5,
"DiskType": "CLOUD_SSD",
"DiskSize": 100,
"LogstashVersion": "6.8.13",
"YMLConfig": "",
"LicenseType": "xpack",
"BindedESInstanceId": "",
"Nodes": [
{
"NodeId": "100003160015134602",
"Ip": "10.0.3.136",
"Port": 9600
},
{
"NodeId": "100003160015134603",
"Ip": "10.0.3.249",
"Port": 9600
},
{
"NodeId": "100003160015279213",
"Ip": "10.0.3.227",
"Port": 9600
},
{
"NodeId": "100003160015279214",
"Ip": "10.0.3.248",
"Port": 9600
},
{
"NodeId": "100003160015279215",
"Ip": "10.0.3.83",
"Port": 9600
}
],
"ExtendedFiles": [
{
"Name": "commons-codec-1.9.jar",
"Size": 1
}
],
"CpuNum": 1,
"MemSize": 1,
"OperationDuration": {
"TimeStart": "02:00",
"TimeZone": "UTC+8",
"Periods": [
1
],
"TimeEnd": "06:00"
},
"TagList": [
{
```

```
"TagKey": "负责人",
"TagValue": "akdfjhs"
}
],
"CreateTime": "2020-09-15 14:29:06",
"UpdateTime": "2020-09-15 15:28:27",
"Deadline": "2017-12-04 00:00:00"
}
],
"RequestId": "5d5a201f-0a3d-485f-a82f-3c73ccxxxxxx"
}
}
```


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.UnAuthorizedOperation	操作未授权。
InternalError	内部错误。
InvalidParameter	参数错误。

重启Logstash实例

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于重启Logstash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：RestartLogstashInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：es-xxxxxxxx
Type	否	Integer	重启类型，0全量重启，1滚动重启 示例值：0

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 重启Logstash实例

用于重启Logstash实例

输入示例

```
https://es.tencentcloudapi.com/?Action=RestartLogstashInstance
&InstanceId=ls-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "7f0e35a7-e03b-42cd-aa4f-0c1471xxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。
ResourceUnavailable	资源不可用。
UnsupportedOperation	操作不支持。

保存并部署管道

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于下发并且部署管道

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：SaveAndDeployLogstashPipeline。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
Pipeline	是	LogstashPipeline	实例管道信息 示例值：{"BatchDelay":1000,"BatchSize":10000}
OpType	否	Integer	操作类型 • 1：只保存 • 2：保存并部署 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 保存并部署管道

输入示例

```
https://es.tencentcloudapi.com/?Action=SaveAndDeployLogstashPipeline
&InstanceId=ls-xxxxxxxx
&Pipeline.PipelineId= pipe_xxx
&Pipeline.PipelineDesc=管道描述
&Pipeline.Config="aW5wdXR7CiAgICAjxxx"
&Pipeline.QueueCheckPointWrites=1000
&Pipeline.QueueType="persisted"
&Pipeline.QueueMaxBytes=1GB
&Pipeline.BatchDelay=50
```

```
&Pipeline.BatchSize=1000
&Pipeline.Workers=10
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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
ResourceInUse	资源被占用。

启动Logstash管道

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于启动Logstash管道

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：StartLogstashPipelines。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
PipelineIds.N	是	Array of String	管道ID列表 示例值：["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 批量部署Logstash管道

输入示例

```
https://es.tencentcloudapi.com/?Action=StartLogstashPipelines
&InstanceId=ls-xxxxxxxx
&PipelineIds.0=pipe_xxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。

停止Logstash管道

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于批量停止Logstash管道

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：StopLogstashPipelines。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
PipelineIds.N	是	Array of String	管道ID列表 示例值：["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 批量停止Logstash管道

输入示例

```
https://es.tencentcloudapi.com/?Action=StopLogstashPipelines
&InstanceId=ls-xxxxxxxx
&PipelineIds.0=pipe_xxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"RequestId": "6eb5c3b2-0bba-4f73-bafb-bd21esxxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。

更新Logstash实例

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

对集群进行节点规格变更，修改实例名称，修改配置，等操作。参数中InstanceId为必传参数，参数传递组合及含义如下：

- InstanceName：修改实例名称(仅用于标识实例)
- NodeNum: 修改实例节点数量（节点横向扩缩容，纵向扩缩容等）
- YMLConfig: 修改实例YML配置
- BindedES：修改绑定的ES集群配置
  以上参数组合只能传递一种，多传或少传均会导致请求失败

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：UpdateLogstashInstance。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
NodeNum	否	Integer	实例节点数量 示例值：2
YMLConfig	否	String	实例YML配置 示例值： {"slowlog.threshold.warn":"2s","slowlog.threshold.info":"1s","slowlog.threshold.
BindedES	否	LogstashBindedES	实例绑定的ES集群信息 示例值：{"ESInstanceId":"es-1ta7zxxx"}
InstanceName	否	String	实例名称 示例值：testnew
ExtendedFiles.N	否	Array of LogstashExtendedFile	扩展文件列表
NodeType	否	String	实例规格 示例值：LOGSTASH.S1.LARGE16
DiskSize	否	Integer	节点磁盘容量 示例值：100
OperationDuration	否	OperationDurationUpdated	可维护时间段

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Logstash实例名称

用以修改指定Logstash实例的名称

输入示例

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

{
"InstanceId": "ls-xxxxxx",
"InstanceName": "newName"
}
```

输出示例

```
{
"Response": {
"RequestId": "c96a110c-7493-452d-a99b-683d07xxxxxx"
}
}
```

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.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
ResourceInUse	资源被占用。
ResourceInUse.Order	订单被锁定。
ResourceInsufficient	资源不足。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
UnsupportedOperation	操作不支持。

版权所有：腾讯云计算（北京）有限责任公司

查询Logstash实例操作记录

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

查询实例指定条件下的操作记录

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：DescribeLogstashInstanceOperations。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-f5mwm28u
StartTime	否	String	起始时间, e.g. "2019-03-07 16:30:39" 示例值：2021-01-05 20:18:03
EndTime	否	String	结束时间, e.g. "2019-03-30 20:18:03" 示例值：2021-01-05 20:28:03
Offset	否	Integer	分页起始值 示例值：0
Limit	否	Integer	分页大小 示例值：30

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	操作记录总数 示例值：1
Operations	Array of Operation	操作记录
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Logstash实例操作记录

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeLogstashInstanceOperations
&InstanceId=ls-f5mwm28u
```

```
&StartTime=2021-01-05 20:18:03
&EndTime=2021-01-05 20:28:03
&Offset=0
&Limit=30
&<公共请求参数>
```

输出示例


```
{
"Response": {
"TotalCount": 1,
"Operations": [
{
"Id": 6173,
"StartTime": "2021-01-05 20:25:39",
"Type": "CreateInstance",
"Detail": {
"OldInfo": [],
"NewInfo": []
},
"Result": "completed",
"Tasks": [
{
"Name": "prepareResource",
"Progress": 1,
"FinishTime": "2021-01-05 20:27:39",
"SubTasks": []
},
{
"Name": "deployESCluster",
"Progress": 1,
"FinishTime": "2019-03-07 16:34:32",
"SubTasks": []
},
{
"Name": "deployKibana",
"Progress": 1,
"FinishTime": "2019-03-07 16:35:13",
"SubTasks": []
},
{
"Name": "configLB",
"Progress": 1,
"FinishTime": "2019-03-07 16:35:15",
"SubTasks": []
}
],
"Progress": 1
}
],
"RequestId": "870dd618-b1ae-40cc-a5a9-22b867367ed7"
}
}
```


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。
ResourceInUse	资源被占用。

获取Logstash实例管道列表

最近更新时间：2025-04-25 01:31:49

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于获取Logstash实例管道列表

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：DescribeLogstashPipelines。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	管道总数 示例值：10
LogstashPipelineList	Array of LogstashPipelineInfo	管道列表 示例值：[{}]
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 获取Logstash实例管道列表

获取Logstash实例管道列表

输入示例

```
https://es.tencentcloudapi.com/?Action=DescribeLogstashPipelines
&InstanceId=ls-xxxxxxxx
&<公共请求参数>
```

输出示例

```
{
"Response": {
"TotalCount": 2,
"LogstashPipelineList": [
```

```
{
"PipelineId": "g4cqkhhr_1",
"PipelineDesc": "企业基本信息",
"Config": "input{\n # xxx}",
"Status": -2,
"Workers": 11,
"BatchSize": 1111,
"BatchDelay": 111,
"QueueType": "memory",
"QueueMaxBytes": "1111MB",
"QueueCheckPointWrites": 1111,
"CreateTime": "2023-12-14 11:09:32",
"UpdateTime": "2023-12-14 11:15:44"
},
{
"PipelineId": "g4cqkhhr_2",
"PipelineDesc": "企业",
"Config": "input{xxx}",
"Status": 2,
"Workers": 11,
"BatchSize": 1111,
"BatchDelay": 111,
"QueueType": "memory",
"QueueMaxBytes": "1111MB",
"QueueCheckPointWrites": 1111,
"CreateTime": "2023-12-14 11:09:32",
"UpdateTime": "2023-12-14 11:15:44"
}
],
"RequestId": ""
}
}
```


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。

更新管道描述

最近更新时间：2025-04-25 01:31:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

用于更新管道描述信息

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：UpdateLogstashPipelineDesc。
Version	是	String	公共参数，本接口取值：2018-04-16。
Region	是	String	公共参数，详见产品支持的 地域列表。
InstanceId	是	String	实例ID 示例值：ls-xxxxxxxx
PipelineId	是	String	实例管道ID 示例值：pipe1
PipelineDesc	是	String	管道描述信息 示例值：test pipe

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 更新实例管道的描述信息

更新实例管道的描述信息

输入示例

```
https://es.tencentcloudapi.com/?Action=UpdateLogstashPipelineDesc
&InstanceId=ls-xxxxxxxx
&PipelineId=pipeId1
&PipelineDesc=testpipe
&<公共请求参数>
```

输出示例

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

```
"RequestId": "7f0e35a7-e03b-42cd-aa4f-0c1471xxxxxx"
}
}
```

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。
ResourceInUse	资源被占用。

数据结构

最近更新时间：2025-06-24 01:24:14

BackingIndexMetaField

后备索引元数据字段

被如下接口引用：DescribeIndexList, DescribeIndexMeta。

名称	类型	描述
IndexName	String	后备索引名 注意：此字段可能返回 null，表示取不到有效值。 示例值：testnew
IndexStatus	String	后备索引状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：GREEN
IndexStorage	Integer	后备索引存储大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：30GB
IndexPhrase	String	后备索引当前生命周期 注意：此字段可能返回 null，表示取不到有效值。 示例值：hot
IndexCreateTime	String	后备索引创建时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2021-03-01 10:00:00

ClusterView

集群维度视图数据

被如下接口引用：DescribeViews。

名称	类型	描述
Health	Float	集群健康状态 示例值：1
Visible	Float	集群是否可见 示例值：1
Break	Float	集群是否熔断 示例值：1
AvgDiskUsage	Float	平均磁盘使用率 示例值：0.7
AvgMemUsage	Float	平均内存使用率 示例值：0.1
AvgCpuUsage	Float	平均cpu使用率 示例值：0.1
TotalDiskSize	Integer	集群总存储大小 示例值：0.1
TargetNodeTypes	Array of String	客户端请求节点 示例值：["hot"]
NodeNum	Integer	在线节点数 示例值：10
TotalNodeNum	Integer	总节点数 示例值：10

版权所有：腾讯云计算（北京）有限责任公司

名称	类型	描述
DataNodeNum	Integer	数据节点数 示例值：7
IndexNum	Integer	索引数 示例值：100
DocNum	Integer	文档数 示例值：100000
DiskUsedInBytes	Integer	磁盘已使用字节数 示例值：10000
ShardNum	Integer	分片个数 示例值：10000
PrimaryShardNum	Integer	主分片个数 示例值：1000
RelocatingShardNum	Integer	迁移中的分片个数 示例值：1
InitializingShardNum	Integer	初始化中的分片个数 示例值：1
UnassignedShardNum	Integer	未分配的分片个数 示例值：0
TotalCosStorage	Integer	企业版COS存储容量大小，单位GB 示例值：1
SearchableSnapshotCosBucket	String	企业版集群可搜索快照cos存放的bucket名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：es-3p7lkxxx-1021
SearchableSnapshotCosAppId	String	企业版集群可搜索快照cos所属appid 注意：此字段可能返回 null，表示取不到有效值。 示例值：1221

CommonIndexInfo

普通索引信息列表

被如下接口引用：DescribeUserCosSnapshotList。

名称	类型	必选	描述
IndexName	String	否	普通索引名 注意：此字段可能返回 null，表示取不到有效值。 示例值：test1
IsShardComplete	Integer	否	分片状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：1

CosBackup

ES cos自动备份信息

被如下接口引用：DescribeInstances, UpdateInstance。

名称	类型	必选	描述
IsAutoBackup	Boolean	是	是否开启cos自动备份 示例值：true
BackupTime	String	是	自动备份执行时间（精确到小时）, e.g. "22:00" 示例值：00:00

名称	类型	必选	描述
EsRepositoryType	Integer	否	0 腾讯云仓库; 1 客户仓库 示例值：0
UserEsRepository	String	否	客户快照仓库名称 示例值：testRepository
StorageDuration	Integer	否	快照存储周期 单位天 示例值：15
AutoBackupInterval	Integer	否	自动备份频率单位小时 示例值：4

CosSnapShotInfo

无

被如下接口引用：DescribeUserCosSnapshotList。

名称	类型	描述
CosBucket	String	cos 桶名 注意：此字段可能返回 null，表示取不到有效值。 示例值：es-ap-shanghai
BasePath	String	base path 注意：此字段可能返回 null，表示取不到有效值。 示例值：es_backup/es-xxxx/
SnapshotName	String	快照名 注意：此字段可能返回 null，表示取不到有效值。 示例值：test_snapshot
State	String	状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：SUCCESS
Version	String	快照版本 注意：此字段可能返回 null，表示取不到有效值。 示例值：7.14.2
CommonIndexArr	Array of CommonIndexInfo	普通[{"DataStreamName":"ilm-history-5","Is索引信息列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"IndexName":"test1","IsShardComplete":1}]
DataStreamArr	Array of DataStreamInfo	自治索引信息列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"DataStreamName":"ilm-his1-5","IsShardComplete":1}, {"DataStreamName":"index1","IsShardComplete":1}]}]

DataStreamInfo

自治索引信息

被如下接口引用：DescribeUserCosSnapshotList。

名称	类型	必选	描述
DataStreamName	String	否	自治索引名 注意：此字段可能返回 null，表示取不到有效值。 示例值：index1
IsShardComplete	Integer	否	分片状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：1

DescribeInstancePluginInfo

插件信息

被如下接口引用：DescribeInstancePluginList。

名称	类型	描述
PluginName	String	插件名 示例值：analysis-qq
PluginVersion	String	插件版本 示例值：6.4.3
PluginDesc	String	插件描述 示例值：插件描述
Status	Integer	插件状态：-2 已卸载 -1 卸载中 0 安装中 1 已安装 示例值：1
Removable	Boolean	插件是否可卸载 示例值：false
PluginType	Integer	0：系统插件 示例值：0
PluginUpdateTime	String	插件变更时间 示例值：2021-03-01 10:00:00

DiData

数据接入信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
DiId	String	数据接入id 注意：此字段可能返回 null，表示取不到有效值。 示例值：sdi-xxxx
CreateTime	String	数据接入创建时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2024-01-08 11:48:18
Status	Integer	数据接入状态，0处理中，1正常，-2删除中，-3已删除 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
DiDataSourceCvm	DiDataSourceCvm	cvm数据源信息 注意：此字段可能返回 null，表示取不到有效值。
DiDataSourceTke	DiDataSourceTke	tke数据源信息 注意：此字段可能返回 null，表示取不到有效值。
DiDataSinkServerless	DiDataSinkServerless	serverless目的端信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"ServerlessId":"index-xxxxx"}
DiDataSourceType	String	数据接入类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：cvm_collector

DiDataSinkServerless

数据接入serverless目的端信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
ServerlessId	String	serverless实例id 注意：此字段可能返回 null，表示取不到有效值。 示例值：index-xxxx

DiDataSourceCvm

数据接入cvm数据源信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
VpcId	String	vpc id 注意：此字段可能返回 null，表示取不到有效值。 示例值：vpc-xxxx
LogPaths	Array of String	采集路径列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[/test, /tmp]
CvmInstances	Array of DiDataSourceCvmInstance	cvm实例信息列表 注意：此字段可能返回 null，表示取不到有效值。
CollectorId	String	采集器id 注意：此字段可能返回 null，表示取不到有效值。 示例值：coll-xxxxxxxx

DiDataSourceCvmInstance

数据接入cvm实例信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
InstanceId	String	cvm实例id 注意：此字段可能返回 null，表示取不到有效值。 示例值：ins-xxxx
VpcId	String	vpc id 注意：此字段可能返回 null，表示取不到有效值。 示例值：vpc-xxxx
SubnetId	String	子网id 注意：此字段可能返回 null，表示取不到有效值。 示例值：subnet-xxxx
ErrMsg	String	错误信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：ok

DiDataSourceTke

数据接入tke数据源信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
VpcId	String	vpc id 注意：此字段可能返回 null，表示取不到有效值。 示例值：vpc-xxxx
TkeId	String	tke实例id 注意：此字段可能返回 null，表示取不到有效值。 示例值：cls-xxxx

版权所有：腾讯云计算（北京）有限责任公司

名称	类型	描述
CollectorId	String	采集器id 注意：此字段可能返回 null，表示取不到有效值。 示例值：coll-xxxxxxxx
CollectorName	String	采集源名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：sdi-xxxx_index-xxxx
CollectorType	String	采集器类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：filebeat
CollectorVersion	String	采集器版本 注意：此字段可能返回 null，表示取不到有效值。 示例值：7.14.0
IncludeNamespaces	Array of String	tke包含的命名空间 注意：此字段可能返回 null，表示取不到有效值。 示例值：["index","apparate-acceleration","default","kube-node-lease","kube-public","kube-system","tcr-assistant-system"]
ExcludeNamespaces	Array of String	tke不包含的命名空间 注意：此字段可能返回 null，表示取不到有效值。 示例值：["index","apparate-acceleration","default","kube-node-lease","kube-public","kube-system","tcr-assistant-system"]
PodLabelKeys	Array of String	tke pod标签名 注意：此字段可能返回 null，表示取不到有效值。 示例值：["es_test"]
PodLabelValues	Array of String	tke pod标签值 注意：此字段可能返回 null，表示取不到有效值。 示例值：["es_test"]
ContainerName	String	tke容器名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：container
ConfigContent	String	tke采集器beat配置 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"input": test}
InputType	String	/ 注意：此字段可能返回 null，表示取不到有效值。 示例值：container
InputPath	String	TKE 日志采集路径 注意：此字段可能返回 null，表示取不到有效值。 示例值：var/log/*

DiSourceCvm

数据接入cvm数据源

被如下接口引用：CreateServerlessInstance。

名称	类型	必选	描述
VpcId	String	否	数据源所属vpc id，创建后不允许修改 示例值：vpc-xxxxxxxx
CvmIds	Array of String	否	cvm列表 示例值：[ins-xxxx, ins-xxxx]
LogPaths	Array of String	否	采集路径列表，支持通配符 示例值：[/test, /tmp]

DiSourceTke

数据接入tke数据源

被如下接口引用：CreateServerlessInstance。

名称	类型	必选	描述
VpcId	String	否	数据源所属vpc id，创建后不允许修改 示例值：vpc-xxxx
TkeId	String	否	tke实例id，创建后不允许修改 示例值：cls-xxxx
IncludeNamespaces	Array of String	否	tke包含的命名空间 示例值：["default"]
ExcludeNamespaces	Array of String	否	tke不包含的命名空间 示例值：["default"]
ContainerName	String	否	tke容器名称 示例值：tke
LogFilters	String	否	tke日志内容过滤 示例值：无使用
ConfigContent	String	否	tke beats配置项 示例值："\nprocessors:\n - decode_json_fields:\n fields: ["message"]\n process_array: false\n max_depth: 1\n target: ""\n overwrite_keys: false\n add_error_key: true\n - drop_fields:\n fields: ["field1", "field2"]\n ignore_missing: false\n"
PodLabel	Array of DiSourceTkePodLabel	否	tke pod标签 示例值：[{"Key":"logCollect","Value":"index-fafafaff"}]
InputType	String	否	/ 示例值：log
InputPath	String	否	tke 日志采集路径 示例值：/var/log/*.log

DiSourceTkePodLabel

tke pod标签

被如下接口引用：CreateServerlessInstance。

名称	类型	必选	描述
Key	String	否	标签key 注意：此字段可能返回 null，表示取不到有效值。 示例值：logCollect
Value	String	否	标签value 注意：此字段可能返回 null，表示取不到有效值。 示例值：index-xxxx

DiagnoseJobMeta

智能运维支持的诊断项和元信息

被如下接口引用：GetDiagnoseSettings。

名称	类型	描述
JobName	String	智能运维诊断项英文名 示例值：cluster_health
JobZhName	String	智能运维诊断项中文名 示例值：集群健康状态诊断

名称	类型	描述
JobDescription	String	智能运维诊断项描述 示例值：诊断集群健康状态

DiagnoseJobResult

智能运维诊断项结果

被如下接口引用：DescribeDiagnose。

名称	类型	描述
JobName	String	诊断项名 示例值：cluster_health
Status	Integer	诊断项状态：-2失败，-1待重试，0运行中，1成功 示例值：0
Score	Integer	诊断项得分 示例值：100
Summary	String	诊断摘要 示例值：配置合理
Advise	String	诊断建议 示例值：保持住
Detail	String	诊断详情 示例值：配置合理
MetricDetails	Array of MetricDetail	诊断指标详情 注意：此字段可能返回 null，表示取不到有效值。
LogDetails	Array of LogDetail	诊断日志详情 注意：此字段可能返回 null，表示取不到有效值。
SettingDetails	Array of SettingDetail	诊断配置详情 注意：此字段可能返回 null，表示取不到有效值。

DiagnoseResult

智能运维诊断结果

被如下接口引用：DescribeDiagnose。

名称	类型	描述
InstanceId	String	ES实例ID 示例值：es-abcdefgh
RequestId	String	诊断报告ID 示例值：ewtd-xcafdg-dsaq
CreateTime	String	诊断触发时间 示例值：2021-03-01 10:00:00
Completed	Boolean	诊断是否完成 示例值：true
Score	Integer	诊断总得分 示例值：100
JobType	Integer	诊断类型，2 定时诊断，3 客户手动触发诊断 示例值：3
JobParam	JobParam	诊断参数，如诊断时间，诊断索引等
JobResults	Array of DiagnoseJobResult	诊断项结果列表

DictInfo

ik插件词典信息

被如下接口引用：DescribeInstances。

名称	类型	必选	描述
Key	String	是	词典键值 示例值：test.dic
Name	String	是	词典名称 示例值：test.dic
Size	Integer	是	词典大小，单位B 示例值：128

Dimension

智能运维指标维度

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Key	String	智能运维指标维度Key 示例值：节点ID
Value	String	智能运维指标维度值 示例值：1232334652

EnableScheduleOperationDuration

置放群组异步任务维护的时间段

被如下接口引用：CreateInstance, DescribeInstances, RestartNodes, UpdateInstance, UpgradeInstance。

名称	类型	必选	描述
Periods	Array of String	否	支持开启异步任务的日期 注意：此字段可能返回 null，表示取不到有效值。 示例值：[2024.09.24, 2024.09.26]
TimeStart	String	否	支持开启异步的开始时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：06:00
TimeEnd	String	否	支持开启异步的结束时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：08:00
TimeZone	String	否	支持开启异步的时区 注意：此字段可能返回 null，表示取不到有效值。 示例值：UTC+8

EsAcl

ES集群配置项

被如下接口引用：DescribeInstances, DescribeServerlessSpaces, UpdateInstance, UpdateServerlessSpace。

名称	类型	必选	描述
BlackIpList	Array of String	否	kibana访问黑名单 示例值：["10...*"]
WhiteIpList	Array of String	否	kibana访问白名单 示例值：["127...*"]

EsConfigSetInfo

配置组信息

被如下接口引用：DescribeInstances, UpdateInstance。

名称	类型	必选	描述
Type	String	是	配置组类型，如ldap,ad等 示例值：ldap
EsConfig	String	是	" {"order":0,"url":"ldap://10.0.1.72:389","bind_dn":"cn=admin,dc=tencent,dc=com","user_search.base_dn":"dc=te (cn={0})","group_search.base_dn":"dc=tencent,dc=com"}" 示例值：参考描述

EsDictionaryInfo

ES 词库信息

被如下接口引用：DescribeInstances。

名称	类型	必选	描述
MainDict	Array of DictInfo	是	启用词词典列表 示例值：[{"Id":0,"UserCosAddress":"https://es-cos-12577xxxx.cos.ap-guangzhou.myqcloud.com/ik_start.dic","Key":"IKPlug/100007799884/es-h687vonp/maindict/temp/ik_start.dic","Name":"ik_start.dic","Size":40,"RemotePath":""}]
Stopwords	Array of DictInfo	是	停用词词典列表 示例值：[{"Id":0,"UserCosAddress":"https://es-cos-125778xxxx.cos.ap-guangzhou.myqcloud.com/ik_stop.dic","Key":"","Name":"ik_stop.dic","Size":116,"RemotePath":""}]
QQDict	Array of DictInfo	是	QQ分词词典列表 示例值：pwd
Synonym	Array of DictInfo	是	同义词词典列表 示例值：[{}]
UpdateType	String	是	更新词典类型 示例值：UpdateIkConfig
AnsjMain	Array of DictInfo	否	ansj启用词词典列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{}]
AnsjStop	Array of DictInfo	否	ansj停用词词典列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{}]
AnsjAmbiguity	Array of DictInfo	否	ansj歧义词库列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{}]
AnsjSynonyms	Array of DictInfo	否	ansj同义词词典列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{}]

EsPublicAcl

ES公网访问控制信息

被如下接口引用：UpdateInstance。

名称	类型	必选	描述
BlackIpList	Array of String	否	访问黑名单

名称	类型	必选	描述
			示例值：["123.207.123.xxx"]
WhiteIpList	Array of String	否	访问白名单 示例值：["123.207.123.253"]

Failures

索引备份失败的数据结构

被如下接口引用：DescribeClusterSnapshot。

名称	类型	必选	描述
Index	String	否	备份失败的索引名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：index_name
ShardId	Integer	否	快照失败的分片号 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
Reason	String	否	快照失败的原因 注意：此字段可能返回 null，表示取不到有效值。 示例值：no such index [gz_test.index_monitor@2025-03-24]
Status	String	否	快照失败的状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：401

IndexMetaField

索引元数据字段

被如下接口引用：DescribeIndexList, DescribeIndexMeta。

名称	类型	描述
IndexType	String	索引类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：auto
IndexName	String	索引名 注意：此字段可能返回 null，表示取不到有效值。 示例值：testnew
IndexMetaJson	String	索引元数据JSON 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"mappings": {"properties":{"@timestamp":{"type":"date"}}}, "settings":{"index.numb
IndexStatus	String	索引状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：GREEN
IndexStorage	Integer	索引存储大小，单位Byte 注意：此字段可能返回 null，表示取不到有效值。 示例值：30
IndexCreateTime	String	索引创建时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2021-03-01 10:00:00
BackingIndices	Array of BackingIndexMetaField	后备索引 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"IndexName":".ds-at-20250324190302-2025.03.24-000001","IndexStatus":"green","IndexStorage":416,"IndexPhrase":"data_hot,data_warm,da
ClusterId	String	索引所属集群ID 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
		示例值：es-abcdefgh
ClusterName	String	索引所属集群名 注意：此字段可能返回 null，表示取不到有效值。 示例值：testnew
ClusterVersion	String	索引所属集群版本 注意：此字段可能返回 null，表示取不到有效值。 示例值：7.14.2
IndexPolicyField	IndexPolicyField	索引生命周期字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"WarmEnable":"false","WarmMinAge":"","ColdEnable":"false","ColdMinAge":"","Col
IndexOptionsField	IndexOptionsField	索引自治字段 注意：此字段可能返回 null，表示取不到有效值。 示例值： {"ExpireMaxAge":"","ExpireMaxSize":"","RolloverMaxAge":"1d","RolloverDynamic":"true","S
IndexSettingsField	IndexSettingsField	索引配置字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"NumberOfShards":"5","RefreshInterval":"30s","CustomSetting":"{"index.number_
IndexAliasesField	Array of String	索引别名字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：[alias1,alias2]
AppId	Integer	索引所属集群APP ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：10004222
IndexDocs	Integer	索引文档数 注意：此字段可能返回 null，表示取不到有效值。 示例值：0

IndexOptionsField

索引自治字段

被如下接口引用：DescribeIndexList, DescribeIndexMeta。

名称	类型	必选	描述
ExpireMaxAge	String	否	过期时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d
ExpireMaxSize	String	否	过期大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：2GB
RolloverMaxAge	String	否	滚动周期 注意：此字段可能返回 null，表示取不到有效值。 示例值：1d
RolloverDynamic	String	否	是否开启动态滚动 注意：此字段可能返回 null，表示取不到有效值。 示例值：true
ShardNumDynamic	String	否	是否开启动态分片 注意：此字段可能返回 null，表示取不到有效值。 示例值：true
TimestampField	String	否	时间分区字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：@timestamp
WriteMode	String	否	写入模式 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	必选	描述
			示例值：append_only

IndexPolicyField

索引生命周期字段

被如下接口引用：DescribeIndexList, DescribeIndexMeta。

名称	类型	必选	描述
WarmEnable	String	否	是否开启warm阶段 注意：此字段可能返回 null，表示取不到有效值。 示例值：false
WarmMinAge	String	否	warm阶段转入时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d
ColdEnable	String	否	是否开启cold阶段 注意：此字段可能返回 null，表示取不到有效值。 示例值：false
ColdMinAge	String	否	cold阶段转入时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d
FrozenEnable	String	否	是否开启frozen阶段 注意：此字段可能返回 null，表示取不到有效值。 示例值：false
FrozenMinAge	String	否	frozen阶段转入时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d
ColdAction	String	否	/ 注意：此字段可能返回 null，表示取不到有效值。 示例值：searchable_snapshot

IndexSettingsField

索引配置字段

被如下接口引用：DescribeIndexList, DescribeIndexMeta。

名称	类型	必选	描述
NumberOfShards	String	否	索引主分片数 注意：此字段可能返回 null，表示取不到有效值。 示例值：5
NumberOfReplicas	String	否	索引副本分片数 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
RefreshInterval	String	否	索引刷新频率 注意：此字段可能返回 null，表示取不到有效值。 示例值：20s

InstanceInfo

实例详细信息

被如下接口引用：DescribeInstances。

名称	类型	描述
InstanceId	String	实例ID 示例值：es-xxxxxxxx
InstanceName	String	实例名称 示例值：实例名称
Region	String	地域 示例值：ap-guangzhou
Zone	String	可用区 示例值：ap-guangzhou-1
AppId	Integer	用户ID 示例值：133xxxxxx
Uin	String	用户UIN 示例值：100xxxxxxxx
VpcUid	String	实例所属VPC的UID 示例值：vpc-xxxxxxx
SubnetUid	String	实例所属子网的UID 示例值：subnet-xxxxxxx
Status	Integer	实例状态，0:处理中,1:正常,-1:停止,-2:销毁中,-3:已销毁, -4:隔离中,2:创建集群时初始化中 示例值：1
RenewFlag	String	自动续费标识。取值范围： RENEW_FLAG_AUTO：自动续费 RENEW_FLAG_MANUAL：不自动续费 默认取值： RENEW_FLAG_DEFAULT：不自动续费 若该参数指定为 RENEW_FLAG_AUTO，在账户余额充足的情况下，实例到期后将按月自动续费。 示例值：NOTIFY_AND_AUTO_RENEW
ChargeType	String	实例计费模式。取值范围： PREPAID：表示预付费，即包年包月 POSTPAID_BY_HOUR：表示后付费，即按量计费 CDHPAID：CDH付费，即只对CDH计费，不对CDH上的实例计费。 示例值：PREPAID
ChargePeriod	Integer	包年包月购买时长,单位:月 示例值：6
NodeType	String	节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：ES.S1.SMALL2
NodeNum	Integer	节点个数 示例值：3
CpuNum	Integer	节点CPU核数 示例值：4
MemSize	Integer	节点内存大小，单位GB 示例值：8
DiskType	String	节点磁盘类型 示例值：CLOUD_PREMIUM

名称	类型	描述
DiskSize	Integer	节点磁盘大小，单位GB 示例值：100
EsDomain	String	ES域名 示例值：http://es-xxxxxxxx
EsVip	String	ES VIP 示例值：1.1.1.1
EsPort	Integer	ES端口 示例值：9200
KibanaUrl	String	Kibana访问url 示例值：http://es-xxxxxxxx
EsVersion	String	ES版本号 示例值：5.6.4
EsConfig	String	ES配置项 示例值："{}"
EsAcl	EsAcl	Kibana访问控制配置
CreateTime	String	实例创建时间 示例值：2021-03-01 10:00:00
UpdateTime	String	实例最后修改操作时间 示例值：2021-03-01 10:00:00
Deadline	String	实例到期时间 示例值：2021-03-01 10:00:00
InstanceType	Integer	实例类型（实例类型标识，当前只有1,2两种） 示例值：1
IkConfig	EsDictionaryInfo	Ik分词器配置
MasterNodeInfo	MasterNodeInfo	专用主节点配置
CosBackup	CosBackup	cos自动备份配置
AllowCosBackup	Boolean	是否允许cos自动备份 示例值：true
TagList	Array of TagInfo	实例拥有的标签列表
LicenseType	String	License类型 • oss：开源版 • basic：基础版 • platinum：白金版 默认值platinum 示例值：platinum
EnableHotWarmMode	Boolean	是否为冷热集群 • true: 冷热集群 • false: 非冷热集群 注意：此字段可能返回 null，表示取不到有效值。 示例值：true
WarmNodeType	String	温节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
		示例值：ES.S1.SMALL2
WarmNodeNum	Integer	温节点个数 注意：此字段可能返回 null，表示取不到有效值。 示例值：3
WarmCpuNum	Integer	温节点CPU核数 注意：此字段可能返回 null，表示取不到有效值。 示例值：2
WarmMemSize	Integer	温节点内存内存大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：4
WarmDiskType	String	温节点磁盘类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：CLOUD_PREMIUM
WarmDiskSize	Integer	温节点磁盘大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：100
NodeInfoList	Array of NodeInfo	集群节点信息列表 注意：此字段可能返回 null，表示取不到有效值。
EsPublicUrl	String	Es公网地址 注意：此字段可能返回 null，表示取不到有效值。 示例值：http://es-xxxxxxxx
MultiZoneInfo	Array of ZoneDetail	多可用区网络信息 注意：此字段可能返回 null，表示取不到有效值。
DeployMode	Integer	部署模式 • 0：单可用区 • 1：多可用区，北京、上海、上海金融、广州、南京、香港、新加坡、法兰克福（白名单控制） 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
PublicAccess	String	ES公网访问状态 • OPEN：开启 • CLOSE：关闭 注意：此字段可能返回 null，表示取不到有效值。 示例值：OPEN
EsPublicAcl	EsAcl	ES公网访问控制配置
KibanaPrivateUrl	String	Kibana内网地址 注意：此字段可能返回 null，表示取不到有效值。 示例值：http://es-xxxxxxxx
KibanaPublicAccess	String	Kibana公网访问状态 • OPEN：开启 • CLOSE：关闭 注意：此字段可能返回 null，表示取不到有效值。 示例值：OPEN
KibanaPrivateAccess	String	Kibana内网访问状态 • OPEN：开启 • CLOSE：关闭 注意：此字段可能返回 null，表示取不到有效值。 示例值：OPEN
SecurityType	Integer	6.8（及以上版本）基础版是否开启xpack security认证 • 1：不开启

名称	类型	描述
		• 2：开启 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
SceneType	Integer	场景化模板类型：0、不开启；1、通用场景；2、日志场景；3、搜索场景 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
KibanaConfig	String	Kibana配置项 注意：此字段可能返回 null，表示取不到有效值。 示例值："{}"
KibanaNodeInfo	KibanaNodeInfo	Kibana节点信息 注意：此字段可能返回 null，表示取不到有效值。
WebNodeTypeInfo	WebNodeTypeInfo	可视化节点配置 注意：此字段可能返回 null，表示取不到有效值。
Jdk	String	JDK类型，oracle或kona 注意：此字段可能返回 null，表示取不到有效值。 示例值：oracle
Protocol	String	集群网络通讯协议 注意：此字段可能返回 null，表示取不到有效值。 示例值：https
SecurityGroups	Array of String	安全组id 注意：此字段可能返回 null，表示取不到有效值。 示例值：xxxabd
ColdNodeType	String	冷节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 注意：此字段可能返回 null，表示取不到有效值。 示例值：ES.S1.SMALL2
ColdNodeNum	Integer	冷节点个数 注意：此字段可能返回 null，表示取不到有效值。 示例值：3
ColdCpuNum	Integer	冷节点CPU核数 注意：此字段可能返回 null，表示取不到有效值。 示例值：2
ColdMemSize	Integer	冷节点内存大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：20
ColdDiskType	String	冷节点磁盘类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：CLOUD_PREMIUM
ColdDiskSize	Integer	冷节点磁盘大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：100
FrozenNodeType	String	冻节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G

名称	类型	描述
		• ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 注意：此字段可能返回 null，表示取不到有效值。 示例值：ES.S1.SMALL2
FrozenNodeNum	Integer	冻节点个数 注意：此字段可能返回 null，表示取不到有效值。 示例值：3
FrozenCpuNum	Integer	冻节点CPU核数 注意：此字段可能返回 null，表示取不到有效值。 示例值：2
FrozenMemSize	Integer	冻节点内存大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：20
FrozenDiskType	String	冻节点磁盘类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：CLOUD_SSD
FrozenDiskSize	Integer	冻节点磁盘大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：100
HealthStatus	Integer	集群健康状态 -1 未知；0 Green; 1 Yellow; 2 Red 注意：此字段可能返回 null，表示取不到有效值。 示例值：-1
EsPrivateUrl	String	https集群内网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://xxx.com
EsPrivateDomain	String	https集群内网域名 注意：此字段可能返回 null，表示取不到有效值。 示例值：xxx.com
EsConfigSets	Array of EsConfigSetInfo	集群的配置组信息 注意：此字段可能返回 null，表示取不到有效值。
OperationDuration	OperationDuration	集群可维护时间段 注意：此字段可能返回 null，表示取不到有效值。
OptionalWebServiceInfos	Array of OptionalWebServiceInfo	web节点列表 注意：此字段可能返回 null，表示取不到有效值。
AutoIndexEnabled	Boolean	自治索引开关 注意：此字段可能返回 null，表示取不到有效值。 示例值：true
EnableHybridStorage	Boolean	是否支持存储计算分离 注意：此字段可能返回 null，表示取不到有效值。 示例值：true
ProcessPercent	Float	流程进度 注意：此字段可能返回 null，表示取不到有效值。 示例值：1.0
KibanaAlteringPublicAccess	String	Kibana的alerting外网告警策略 • OPEN：开启 • CLOSE：关闭 注意：此字段可能返回 null，表示取不到有效值。 示例值：OPEN
HasKernelUpgrade	Boolean	本月是否有内核可以更新：false-无，true-有 注意：此字段可能返回 null，表示取不到有效值。 示例值：false

名称	类型	描述
CdcId	String	cdcId，使用cdc子网时传递 注意：此字段可能返回 null，表示取不到有效值。 示例值：cluster-xxxxxx
KibanaPrivateVip	String	kibana内网vip 注意：此字段可能返回 null，表示取不到有效值。 示例值：10.0.0.1
CustomKibanaPrivateUrl	String	自定义kibana内网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：http://xxx.com
OutboundPublicAcls	Array of OutboundPublicAcl	节点出站访问详细信息 注意：此字段可能返回 null，表示取不到有效值。
NetConnectScheme	String	网络连接方案 注意：此字段可能返回 null，表示取不到有效值。 示例值：PrivateLink
DisasterRecoverGroupAffinity	Integer	置放群组相关参数 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
SubProductCode	String	子产品ID枚举值： 开源版："sp_es_io2"， 基础版："sp_es_basic"，白金版："sp_es_platinum"，企业版："sp_es_enterprise"，CDC白金版："sp_es_cdc_platinum"，日志增强版："sp_es_enlogging"，tsearch："sp_tsearch_io2"，logstash："sp_es_logstash"，可以为空，为空的时候后台取LicenseType映射该字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：sp_es_platinum
CosBucketStorageSize	Integer	存算分离cos用量，单位M 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
ReadWriteMode	Integer	读写分离模式：0-不开启，1-本地读写分离，2-远端读写分离 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
EnableScheduleRecoverGroup	Boolean	是否有置放群组异步调度任务 注意：此字段可能返回 null，表示取不到有效值。 示例值：false
EnableScheduleOperationDuration	EnableScheduleOperationDuration	异步调度任务的时间 注意：此字段可能返回 null，表示取不到有效值。
EnableDestroyProtection	String	开启集群保护：OPEN-开启，CLOSE-关闭 注意：此字段可能返回 null，表示取不到有效值。 示例值：CLOSE

InstanceLog

ES集群日志详细信息

被如下接口引用：DescribeInstanceLogs, DescribeLogstashInstanceLogs。

名称	类型	描述
Time	String	日志时间 示例值：2025-03-24T19:36:32.240+08:00
Level	String	日志级别 示例值：ERROR
Ip	String	集群节点ip 示例值：9.20.85.138

名称	类型	描述
Message	String	日志内容 示例值：[o.e.x.w.i.s.ExecutableSearchInput] [167075xxx
NodeID	String	集群节点ID 示例值：1670750203001429xxx

JobParam

智能运维诊断参数

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Jobs	Array of String	诊断项列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：[cluster_health]
Indices	String	诊断索引 示例值："*"
Interval	Integer	历史诊断时间 示例值：60

KeyValue

OperationDetail使用此结构的数组描述新旧配置

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
Key	String	键 示例值：WarmDataNodeDiskCount
Value	String	值 示例值：2

KibanaNodeInfo

实例Kibana节点相关信息

被如下接口引用：DescribeInstances。

名称	类型	描述
KibanaNodeType	String	Kibana节点规格 示例值：ES.S1.SMALL2
KibanaNodeNum	Integer	Kibana节点个数 示例值：1
KibanaNodeCpuNum	Integer	Kibana节点CPU数 示例值：2
KibanaNodeMemSize	Integer	Kibana节点内存GB 示例值：4
KibanaNodeDiskType	String	Kibana节点磁盘类型 示例值：CLOUD_PREMIUM
KibanaNodeDiskSize	Integer	Kibana节点磁盘大小 示例值：50

KibanaPublicAcl

kibana公网域名白名单参数

被如下接口引用：UpdateServerlessInstance。

名称	类型	必选	描述
WhiteIpList	Array of String	否	kibana访问白名单 示例值：["123.1.2.4"]

KibanaView

Kibana视图数据

被如下接口引用：DescribeViews。

名称	类型	描述
Ip	String	Kibana节点IP 示例值：9.15.123.xxx
DiskSize	Integer	节点总磁盘大小 示例值：50
DiskUsage	Float	磁盘使用率 示例值：0.1967
MemSize	Integer	节点内存大小 示例值：2
MemUsage	Float	内存使用率 示例值：0
CpuNum	Integer	节点cpu个数 示例值：1
CpuUsage	Float	cpu使用率 示例值：0.029166666666667
Zone	String	可用区 示例值：ap-beijing-7
NodeId	String	ts-0noqayxu-az6-hot-03222010-0 示例值：1658386193000764xxx

LocalDiskInfo

节点本地盘信息

被如下接口引用：CreateInstance, DescribeInstances, UpdateInstance。

名称	类型	必选	描述
LocalDiskType	String	是	本地盘类型 • LOCAL_SATA：大数据型 • NVME_SSD：高IO型 示例值：NVME_SSD
LocalDiskSize	Integer	是	本地盘单盘大小 示例值：7311360
LocalDiskCount	Integer	是	本地盘块数 示例值：1

LogDetail

智能运维日志详情

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Key	String	日志异常名 示例值：BreakException
Advise	String	日志异常处理建议 示例值：建议处理熔断
Count	Integer	日志异常名出现次数 示例值：5

LogstashBindedES

Logstash绑定的ES集群信息

被如下接口引用：UpdateLogstashInstance。

名称	类型	必选	描述
ESInstanceId	String	是	ES集群ID 示例值：es-xxxxxxxx
ESUserName	String	是	ES集群用户名 示例值：elastic
ESPassword	String	是	ES集群密码 示例值：G7j#kL9****

LogstashExtendedFile

Logstash扩展文件信息

被如下接口引用：DescribeLogstashInstances, UpdateLogstashInstance。

名称	类型	必选	描述
Name	String	是	扩展文件名称 示例值：test.conf
Size	Integer	是	扩展文件大小，单位B 示例值：1000

LogstashInstanceInfo

Logstash实例详细信息

被如下接口引用：DescribeLogstashInstances。

名称	类型	描述
InstanceId	String	实例ID 示例值：es-xxxx
InstanceName	String	实例名称 示例值：测试集群
Region	String	地域 示例值：ap-guangzhou
Zone	String	可用区 示例值：ap-guangzhou-1
AppId	Integer	用户ID 示例值：12*488
Uin	String	用户UIN 示例值：100**638

名称	类型	描述
VpcId	String	实例所属VPC的ID 示例值：vpc-xx
SubnetId	String	实例所属子网的ID 示例值：subnet-xx
Status	Integer	实例状态，0:处理中,1:正常,-1停止,-2:销毁中,-3:已销毁 示例值：0
ChargeType	String	实例计费模式。取值范围： PREPAID：表示预付费，即包年包月 POSTPAID_BY_HOUR：表示后付费，即按量计费 CDHPAID：CDH付费，即只对CDH计费，不对CDH上的实例计费。 示例值：PREPAID
ChargePeriod	Integer	包年包月购买时长,单位:月 示例值：1
RenewFlag	String	自动续费标识。取值范围： NOTIFY_AND_AUTO_RENEW：通知过期且自动续费 NOTIFY_AND_MANUAL_RENEW：通知过期不自动续费 DISABLE_NOTIFY_AND_MANUAL_RENEW：不通知过期不自动续费 默认取值：NOTIFY_AND_AUTO_RENEW。若该参数指定为NOTIFY_AND_AUTO_RENEW，在账户余额充足的情况下，实例到期后将按月自动续费。 示例值：NOTIFY_AND_MANUAL_RENEW
NodeType	String	节点规格 • LOGSTASH.S1.SMALL2：1核2G • LOGSTASH.S1.MEDIUM4：2核4G • LOGSTASH.S1.MEDIUM8：2核8G • LOGSTASH.S1.LARGE16：4核16G • LOGSTASH.S1.2XLARGE32：8核32G • LOGSTASH.S1.4XLARGE32：16核32G • LOGSTASH.S1.4XLARGE64：16核64G 示例值：LOGSTASH.S1.SMALL2
NodeNum	Integer	节点个数 示例值：1
DiskType	String	节点磁盘类型 示例值：SSD
DiskSize	Integer	节点磁盘大小，单位GB 示例值：100
LogstashVersion	String	Logstash版本号 示例值：6.8.13
LicenseType	String	License类型 • oss：开源版 • xpack：基础版 默认值xpack 示例值：oss
CreateTime	String	实例创建时间 示例值：2022-10-12 11:25:37
UpdateTime	String	实例最后修改操作时间 示例值：2022-10-12 11:25:37
Deadline	String	实例到期时间 示例值：2017-12-04 00:00:00
Nodes	Array of LogstashNodeInfo	实例节点类型
BindedESInstanceId	String	实例绑定的ES集群ID 示例值：es-hi*8k
YMLConfig	String	实例的YML配置 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
		示例值：{"slowlog.threshold.warn":"3s"}
ExtendedFiles	Array of LogstashExtendedFile	扩展文件列表 注意：此字段可能返回 null，表示取不到有效值。
OperationDuration	OperationDuration	可维护时间段 注意：此字段可能返回 null，表示取不到有效值。
CpuNum	Integer	CPU数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
TagList	Array of TagInfo	实例标签信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"TagKey":"自动化","TagValue":"自动化"}]
MemSize	Integer	内存大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
DeployMode	Integer	部署模式，0：单可用区、1：多可用区 示例值：0
MultiZoneInfo	Array of ZoneDetail	多可用区部署时可用区的详细信息

LogstashNodeInfo

Logstash节点信息

被如下接口引用：DescribeLogstashInstances。

名称	类型	描述
NodeId	String	节点ID 示例值：10616***7299
Ip	String	节点IP 示例值：127...*
Port	Integer	节点端口 示例值：8080
Zone	String	节点所在zone 示例值：ap-beijing-1

LogstashPipeline

Logstash管道信息

被如下接口引用：SaveAndDeployLogstashPipeline。

名称	类型	必选	描述
PipelineId	String	否	管道ID 示例值：pipe_test
PipelineDesc	String	否	管道描述信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：test_pipe
Config	String	否	管道配置内容 示例值：input { elasticsearch { user => "elastic"} } output { elasticsearch { user => "elastic" } }
Workers	Integer	否	管道的Worker数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：1

名称	类型	必选	描述
BatchSize	Integer	否	管道批处理大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：1000
BatchDelay	Integer	否	管道批处理延迟 注意：此字段可能返回 null，表示取不到有效值。 示例值：50
QueueType	String	否	管道缓冲队列类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：memory
QueueMaxBytes	String	否	管道缓冲队列大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：1GB
QueueCheckPointWrites	Integer	否	管道缓冲队列检查点写入数 注意：此字段可能返回 null，表示取不到有效值。 示例值：1000

LogstashPipelineInfo

Logstash管道信息

被如下接口引用：DescribeLogstashPipelines。

名称	类型	描述
PipelineId	String	管道ID 示例值：6823pipeline
PipelineDesc	String	管道描述信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：日志中有warn
Config	String	管道配置内容 示例值：input { beats { port => "8080" } } output { stdout { codec => debug } }
Status	Integer	管道状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：2
Workers	Integer	管道的Worker数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：4
BatchSize	Integer	管道批处理大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：125
BatchDelay	Integer	管道批处理延迟 注意：此字段可能返回 null，表示取不到有效值。 示例值：50
QueueType	String	管道缓冲队列类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：memory
QueueMaxBytes	String	管道缓冲队列大小 注意：此字段可能返回 null，表示取不到有效值。 示例值：2048
QueueCheckPointWrites	Integer	管道缓冲队列检查点写入数 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
CreateTime	String	创建时间 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
		示例值：2024-04-09 14:53:41
UpdateTime	String	更新时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2024-04-09 14:54:05

MasterNodeInfo

实例专用主节点相关信息

被如下接口引用：DescribeInstances。

名称	类型	描述
EnableDedicatedMaster	Boolean	是否启用了专用主节点 示例值：true
MasterNodeType	String	专用主节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：dedicatedMaster
MasterNodeNum	Integer	专用主节点个数 示例值：1
MasterNodeCpuNum	Integer	专用主节点CPU核数 示例值：8
MasterNodeMemSize	Integer	专用主节点内存大小，单位GB 示例值：16384
MasterNodeDiskSize	Integer	专用主节点磁盘大小，单位GB 示例值：51200
MasterNodeDiskType	String	专用主节点磁盘类型 示例值：CLOUD_SSD

Metric

智能运维指标

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Dimensions	Array of Dimension	指标维度族
Value	Float	指标值 示例值：0.1

MetricAllData

全部指标数据

被如下接口引用：DescribeServerlessMetrics。

名称	类型	描述
IndexTraffic	Float	索引流量 示例值：12.34454

名称	类型	描述
Storage	Float	存储大小 示例值：12.3432
ReadReqTimes	Integer	读请求次数 示例值：21
WriteReqTimes	Integer	写请求次数 示例值：32
DocCount	Integer	文档数量 示例值：32

MetricDetail

智能运维指标详情

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Key	String	指标详情名 示例值：当前熔断节点
Metrics	Array of Metric	指标详情值

MetricMapByIndexId

指标数据map

被如下接口引用：DescribeServerlessMetrics。

名称	类型	描述
IndexId	String	实例id 示例值：index-xxxxxxxx
MetricAllData	MetricAllData	指标数据 示例值： {"IndexTraffic":12302.06,"Storage":43044.33,"ReadReqTimes":2,"WriteReqTimes":111,"DocCount":0}

NodeInfo

集群中一种节点类型（如热数据节点，冷数据节点，专用主节点等）的规格描述信息，包括节点类型，节点个数，节点规格，磁盘类型，磁盘大小等, Type不指定时默认为热数据节点；如果节点为master节点，则DiskType和DiskSize参数会被忽略（主节点无数据盘）

被如下接口引用：CreateInstance, DescribeInstances, UpdateInstance。

名称	类型	必选	描述
NodeNum	Integer	是	节点数量 示例值：3
NodeType	String	是	节点规格 • ES.S1.SMALL2：1核2G • ES.S1.MEDIUM4：2核4G • ES.S1.MEDIUM8：2核8G • ES.S1.LARGE16：4核16G • ES.S1.2XLARGE32：8核32G • ES.S1.4XLARGE32：16核32G • ES.S1.4XLARGE64：16核64G 示例值：ES.S1.4XLARGE64
Type	String	否	节点类型 • hotData: 热数据节点 • warmData: 冷数据节点

名称	类型	必选	描述
			• dedicatedMaster: 专用主节点 默认值为hotData 示例值：dedicatedMaster
DiskType	String	否	节点磁盘类型 • CLOUD_SSD：SSD云硬盘 • CLOUD_PREMIUM：高硬能云硬盘 默认值CLOUD_SSD 示例值：CLOUD_SSD
DiskSize	Integer	否	节点磁盘容量（单位GB） 示例值：50
LocalDiskInfo	LocalDiskInfo	否	节点本地盘信息 注意：此字段可能返回 null，表示取不到有效值。
DiskCount	Integer	否	节点磁盘块数 示例值：1
DiskEncrypt	Integer	否	节点磁盘是否加密 0: 不加密，1: 加密；默认不加密 示例值：0
CpuNum	Integer	否	cpu数目 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
MemSize	Integer	否	内存大小，单位GB 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
DiskEnhance	Integer	否	/ 注意：此字段可能返回 null，表示取不到有效值。 示例值：0

NodeView

节点维度视图数据

被如下接口引用：DescribeViews。

名称	类型	描述
NodeId	String	节点ID 示例值：NodeId
NodeIp	String	节点IP 示例值：NodeIp
Visible	Float	节点是否可见 示例值：Visible
Break	Float	是否熔断 示例值：Break
DiskSize	Integer	节点总磁盘大小 示例值：DiskSize
DiskUsage	Float	磁盘使用率 示例值：DiskUsage
MemSize	Integer	节点内存大小，单位GB 示例值：MemSize
MemUsage	Float	内存使用率 示例值：MemUsage
CpuNum	Integer	节点cpu个数 示例值：CpuNum

名称	类型	描述
CpuUsage	Float	cpu使用率 示例值：CpuUsage
Zone	String	可用区 示例值：Zone
NodeRole	String	节点角色 示例值：NodeRole
NodeHttpIp	String	节点HTTP IP 示例值：NodeHttpIp
JvmMemUsage	Float	JVM内存使用率 示例值：JvmMemUsage
ShardNum	Integer	节点分片数 示例值：ShardNum
DiskIds	Array of String	节点上磁盘ID列表 示例值：DiskIds
Hidden	Boolean	是否为隐藏可用区 示例值：Hidden
IsCoordinationNode	Boolean	是否充当协调节点的角色 示例值：false
CVMStatus	String	CVM运行状态 示例值：RUNNING
CVMDisasterRecoverGroupId	String	cvm绑定的置放群组的id 注意：此字段可能返回 null，表示取不到有效值。 示例值：xxxabd
CVMDisasterRecoverGroupStatus	Integer	cvm绑定置放群组的状态。2: 已绑定；1: 绑定中；0: 未开启 注意：此字段可能返回 null，表示取不到有效值。 示例值：0

Operation

ES集群操作详细信息

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
Id	Integer	操作唯一id 示例值：190587
StartTime	String	操作开始时间 示例值：2023-03-06 16:24:44
Type	String	操作类型 示例值：UpdateKibanaConfig
Detail	OperationDetail	操作详情 示例值：{"OldInfo":[{"Key":"KibanaConfig","Value":"{}"}],"NewInfo":[{"Key":"KibanaConfig","Value":"{"xpack.reporting.csv.maxSizeBytes":"104857600"}"}]}
Result	String	操作结果 示例值：completed
Tasks	Array of TaskDetail	流程任务信息 示例值：[{"Name":"upgradeCheck","Progress":1,"FinishTime":"2020-04-26 14:45:55","SubTasks":[{"Name":"userAgentPipelinesCheck","Type":"cluster","Level":1,"Result":false,"Status":1,"FinishTime":04-26 14:45:51","ErrMsg":"","FailedIndices":[]}]
Progress	Float	操作进度 示例值：1

名称	类型	描述
RollbackTag	Integer	回滚标记， 0未回滚 ，1回滚中，2已回滚 示例值：0
SubAccountUin	String	操作者Uin 注意：此字段可能返回 null，表示取不到有效值。 示例值：1000xxxx

OperationDetail

操作详情

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
OldInfo	Array of KeyValue	实例原始配置信息
NewInfo	Array of KeyValue	实例更新后配置信息

OperationDuration

集群可运维时间

被如下接口引用：CreateInstance, CreateLogstashInstance, DescribeInstances, DescribeLogstashInstances。

名称	类型	必选	描述
Periods	Array of Integer	是	维护周期，表示周一到周日，可取值[0, 6] 注意：此字段可能返回 null，表示取不到有效值。 示例值：[0,1,2]
TimeStart	String	是	维护开始时间 示例值：02:00
TimeEnd	String	是	维护结束时间 示例值：06:00
TimeZone	String	是	时区，以UTC形式表示 示例值：UTC+8

OperationDurationUpdated

集群可运维时间

被如下接口引用：UpdateInstance, UpdateLogstashInstance。

名称	类型	必选	描述
Periods	Array of Integer	是	维护周期，表示周一到周日，可取值[0, 6] 示例值：[0,1,2]
TimeStart	String	是	维护开始时间 示例值：02:00
TimeEnd	String	是	维护结束时间 示例值：06:00
TimeZone	String	是	时区，以UTC形式表示 示例值：UTC+8
MoreInstances	Array of String	否	ES集群ID数组 示例值：[es-c3tz27ww]

OptionalWebServiceInfo

可选web组件信息

被如下接口引用：DescribeInstances。

名称	类型	描述
Type	String	类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：cerebro
Status	Integer	状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
PublicUrl	String	公网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://es-xxxxxxxx.cerebro.tencentelasticsearch.com:9000
PrivateUrl	String	内网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：http://es-xxxxxxxx.interna.cerebro.tencentelasticsearch.com:9000
PublicAccess	String	公网访问权限 注意：此字段可能返回 null，表示取不到有效值。 示例值：OPEN
PrivateAccess	String	内网访问权限 注意：此字段可能返回 null，表示取不到有效值。 示例值：CLOSE
Version	String	版本号 注意：此字段可能返回 null，表示取不到有效值。 示例值：0.9.4
PrivateVip	String	web服务内网vip 注意：此字段可能返回 null，表示取不到有效值。 示例值：10.0.0.1
CustomPrivateUrl	String	自定义cerebro内网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：http://xxxxx.com:9000

OutboundPublicAcl

节点出站访问信息

被如下接口引用：DescribeInstances, UpdateInstance。

名称	类型	必选	描述
NodeType	String	是	允许节点出站访问的节点类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：ES.S1.8XLARGE128
WhiteHostList	Array of String	否	允许节点出站访问的白名单 注意：此字段可能返回 null，表示取不到有效值。 示例值：[""]

ProcessDetail

任务进度详情

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
Completed	Integer	已完成数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：2

名称	类型	描述
Remain	Integer	剩余数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：2
Total	Integer	总数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：4
TaskType	Integer	任务类型： 60：重启型任务 70：分片迁移型任务 80：节点变配任务 注意：此字段可能返回 null，表示取不到有效值。 示例值：60

ServerlessDi

创建serverless索引时创建数据接入

被如下接口引用：CreateServerlessInstance。

名称	类型	必选	描述
DiSourceType	String	否	数据链路采集源类型，如cvm_collector、tke_collector 示例值：cvm_collector
DiSourceCvm	DiSourceCvm	否	cvm数据源
DiSourceTke	DiSourceTke	否	tke数据源

ServerlessIndexMetaField

索引元数据字段

被如下接口引用：DescribeServerlessInstances。

名称	类型	描述
AppId	Integer	索引所属集群APP ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：10004222
IndexName	String	索引名 注意：此字段可能返回 null，表示取不到有效值。 示例值：log-test
IndexMetaJson	String	索引元数据JSON 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"mappings":{"properties":{"filed1":{"analyzer":"custom_analyzer","type":"text"}}}}
IndexDocs	Integer	索引文档数 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
IndexStorage	Integer	索引存储大小，单位Byte 注意：此字段可能返回 null，表示取不到有效值。 示例值：30
IndexCreateTime	String	索引创建时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2021-03-01 10:00:00
InstanceId	String	索引实例ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：index-abcdefgh

名称	类型	描述
IndexOptionsField	ServerlessIndexOptionsField	索引自治字段 注意：此字段可能返回 null，表示取不到有效值。
IndexSettingsField	ServerlessIndexSettingsField	索引配置字段 注意：此字段可能返回 null，表示取不到有效值。
IndexNetworkField	ServerlessIndexNetworkField	索引所属连接相关信息 注意：此字段可能返回 null，表示取不到有效值。
KibanaUrl	String	Kibana公网域名 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://space-xxxxxx.ap-nanjing.kibana.qcloudes.com:5601
KibanaPrivateUrl	String	Kibana内网域名 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://space-xxxxxx-internal.ap-nanjing.kibana.qcloudes.com:5601
IndexAccessUrl	String	索引内网访问地址 注意：此字段可能返回 null，表示取不到有效值。 示例值：space-xxxxxx.ap-nanjing.qcloudes.com
Status	Integer	状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
SpaceId	String	索引空间ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：space-xxx
SpaceName	String	索引空间名 注意：此字段可能返回 null，表示取不到有效值。 示例值：test1
StorageType	Integer	存储类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
TagList	Array of TagInfo	标签信息
IndexTraffic	Float	索引流量，单位byte 注意：此字段可能返回 null，表示取不到有效值。 示例值：3782478.47

ServerlessIndexNetworkField

Serverless实例，网络、索引、kibana等连接信息

被如下接口引用：DescribeServerlessInstances。

名称	类型	描述
Region	String	地域 注意：此字段可能返回 null，表示取不到有效值。 示例值：ap-guangzhou
Zone	String	区域 注意：此字段可能返回 null，表示取不到有效值。 示例值：ap-guangzhou-1
VpcUid	String	vpc唯一ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：vpc-abcdef
SubnetUid	String	子网唯一ID 注意：此字段可能返回 null，表示取不到有效值。 示例值：sub-abcdef
Username	String	用户名 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	描述
		示例值：tom
Password	String	密码 注意：此字段可能返回 null，表示取不到有效值。 示例值：abcd1234

ServerlessIndexOptionsField

索引自治字段

被如下接口引用：DescribeServerlessInstances。

名称	类型	必选	描述
ExpireMaxAge	String	否	过期时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d
TimestampField	String	否	时间分区字段 注意：此字段可能返回 null，表示取不到有效值。 示例值：@timestamp
SinkCycleAge	String	否	标准存储时长 注意：此字段可能返回 null，表示取不到有效值。 示例值：7d
StandardStorageAge	String	否	标准存储时长 注意：此字段可能返回 null，表示取不到有效值。 示例值：3d

ServerlessIndexSettingsField

索引配置字段

被如下接口引用：DescribeServerlessInstances。

名称	类型	必选	描述
NumberOfShards	String	否	索引主分片数 注意：此字段可能返回 null，表示取不到有效值。 示例值：5
RefreshInterval	String	否	索引刷新频率 注意：此字段可能返回 null，表示取不到有效值。 示例值：20s
CustomSetting	String	否	自定义参数 示例值：5

ServerlessSpace

Serverless索引空间信息

被如下接口引用：DescribeServerlessSpaces。

名称	类型	描述
SpaceId	String	Serverless索引空间ID 示例值：space-xxxx
SpaceName	String	Serverless索引空间名 示例值：spaceName
Status	Integer	Serverless索引空间状态，0正常，-1已删除 示例值：1

名称	类型	描述
CreateTime	String	创建日期 示例值：2022-01-01 11:00:00
IndexCount	Integer	空间内索引数量 示例值：1
KibanaUrl	String	kibana公网uri 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://space-xxxxxx.ap-nanjing.kibana.qcloudes.com:5601
KibanaPrivateUrl	String	kibana内网url 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://space-xxxxxx-internal.ap-nanjing.kibana.qcloudes.com:5601
IndexAccessUrl	String	空间内网访问地址 注意：此字段可能返回 null，表示取不到有效值。 示例值：space-xxxxxx.ap-nanjing.qcloudes.com
KibanaPublicAcl	EsAcl	空间白名单 注意：此字段可能返回 null，表示取不到有效值。 示例值：{"WhiteIpList":["127.0.0.1"],"BlackIpList":[]}
KibanaEmbedUrl	String	空间检索分析域名 注意：此字段可能返回 null，表示取不到有效值。 示例值：https://space-xxxxxx.kibana-embed.qcloudes.com:5601
DiDataList	DiData	数据联路 注意：此字段可能返回 null，表示取不到有效值。
VpcInfo	Array of VpcInfo	空间vpc信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{"VpcId":"vpc-xxxxxxx","SubnetId":"subnet-xxxxxxx","VpcUid":11111111,"SubnetUid":11111111,"AvailableIpAddressCount":0}]
Region	String	地域 注意：此字段可能返回 null，表示取不到有效值。 示例值：ap-shanghai
Zone	String	可用区 注意：此字段可能返回 null，表示取不到有效值。 示例值：ap-shanghai-3
EnableKibanaPublicAccess	Integer	kibana公网开关，0关闭，1开启 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
EnableKibanaPrivateAccess	Integer	kibana内网开关，0关闭，1开启 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
AppId	Integer	空间所属appid 注意：此字段可能返回 null，表示取不到有效值。 示例值：10004222
KibanaLanguage	String	//默认en， 可选zh-CN 注意：此字段可能返回 null，表示取不到有效值。 示例值：en
ClusterType	Integer	0 注意：此字段可能返回 null，表示取不到有效值。 示例值：0:普通集群；1:T-Search集群
TagList	Array of TagInfo	空间标签信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：key:value

ServerlessSpaceUser

ServerlessSpaceUser

被如下接口引用：DescribeServerlessSpaceUser。

名称	类型	必选	描述
Username	String	否	用户名 注意：此字段可能返回 null，表示取不到有效值。 示例值：admin
Password	String	否	用户密码 注意：此字段可能返回 null，表示取不到有效值。 示例值：aefafaefax
CreateTime	String	否	创建时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2023-04-25 15:55:31
Status	Integer	否	用户状态 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
IndicesScope	Array of String	否	有权限的索引数组 注意：此字段可能返回 null，表示取不到有效值。 示例值：["index1"]
PrivilegeType	Integer	否	权限类型 注意：此字段可能返回 null，表示取不到有效值。 示例值：0

SettingDetail

智能运维集群配置详情

被如下接口引用：DescribeDiagnose。

名称	类型	描述
Key	String	配置key 示例值：settings.a
Value	String	配置当前值 示例值：value.s
Advise	String	配置处理建议 示例值：建议修改配置

Snapshots

集群快照数据结构

被如下接口引用：DescribeClusterSnapshot。

名称	类型	必选	描述
SnapshotName	String	否	快照名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：snapshot-abc12
Uuid	String	否	快照Uuid 注意：此字段可能返回 null，表示取不到有效值。 示例值：4pf6R*a6DA
Repository	String	否	仓库名称 注意：此字段可能返回 null，表示取不到有效值。 示例值：testRepository
Version	String	否	该快照所属集群的版本号 注意：此字段可能返回 null，表示取不到有效值。

名称	类型	必选	描述
			示例值：7.14.2
Indices	Array of String	否	备份的索引列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：["jmeter_index"]
DataStreams	Array of String	否	备份的datastream列表 注意：此字段可能返回 null，表示取不到有效值。 示例值：["ilm-history-10"]
State	String	否	备份的状态 FAILED 备份失败 IN_PROGRESS 备份执行中 PARTIAL 备份部分成功，部分失败，备份失败的索引和原因会在Failures字段中展示 SUCCESS 备份成功 注意：此字段可能返回 null，表示取不到有效值。 示例值：SUCCESS
StartTime	String	否	快照备份的开始时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2025-03-19T18:02:27.977Z
EndTime	String	否	快照备份的结束时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：2025-03-19T18:02:32.378Z
DurationInMillis	Integer	否	快照备份的耗时时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：4401
TotalShards	Integer	否	备份的总分片数 注意：此字段可能返回 null，表示取不到有效值。 示例值：10
FailedShards	Integer	否	备份失败的分片数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：0
SuccessfulShards	Integer	否	备份成功的分片数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：10
Failures	Array of Failures	否	备份失败的索引分片和失败原因 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{...}]
UserBackUp	String	否	是否用户备份 注意：此字段可能返回 null，表示取不到有效值。 示例值：true

SubTaskDetail

实例操作记录流程任务中的子任务信息（如升级检查任务中的各个检查项）

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
Name	String	子任务名 示例值：shardLimitCheck
Result	Boolean	子任务结果 示例值：true

名称	类型	描述
ErrMsg	String	子任务错误信息 示例值：shards exceeded limit
Type	String	子任务类型 示例值：cluster
Status	Integer	子任务状态，0处理中 1成功 -1失败 示例值：10
FailedIndices	Array of String	升级检查失败的索引名 示例值：jemter_index
FinishTime	String	子任务结束时间 示例值：2025-03-20T07:57:37.798Z
Level	Integer	子任务等级，1警告 2失败 示例值：10

TagInfo

实例标签信息

被如下接口引用：CreateInstance, CreateLogstashInstance, CreateServerlessInstance, CreateServerlessSpaceV2, DescribeInstances, DescribeLogstashInstances, DescribeServerlessInstances, DescribeServerlessSpaces。

名称	类型	必选	描述
TagKey	String	是	标签键 示例值：负责人
TagValue	String	是	标签值 示例值：jack

TaskDetail

实例操作记录中的流程任务信息

被如下接口引用：DescribeInstanceOperations, DescribeLogstashInstanceOperations。

名称	类型	描述
Name	String	任务名 示例值：prepareResource
Progress	Float	任务进度 示例值：1.2
FinishTime	String	任务完成时间 示例值：2023-02-01 12:49:37
SubTasks	Array of SubTaskDetail	子任务
ElapsedTime	Integer	任务花费时间 注意：此字段可能返回 null，表示取不到有效值。 示例值：1
ProcessInfo	ProcessDetail	任务进度详情 注意：此字段可能返回 null，表示取不到有效值。

VpcInfo

vpc信息

被如下接口引用：CreateServerlessSpaceV2, DescribeServerlessSpaces。

名称	类型	必选	描述
VpcId	String	否	vpcId信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：vpc-xxxxxxx
SubnetId	String	否	SubnetId信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：subnet-fafafag
VpcUid	Integer	否	VpcUid信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：8808090
SubnetUid	Integer	否	SubnetUid 信息 注意：此字段可能返回 null，表示取不到有效值。 示例值：8808090
AvailableIpAddressCount	Integer	否	可用ip数量 注意：此字段可能返回 null，表示取不到有效值。 示例值：1

WebNodeTypeInfo

可视化节点配置

被如下接口引用：CreateInstance, DescribeInstances, UpdateInstance。

名称	类型	必选	描述
NodeNum	Integer	是	可视化节点个数，固定为1 示例值：1
NodeType	String	是	可视化节点规格 示例值：ES.S1.SMALL2

ZoneDetail

多可用区部署时可用区的详细信息

被如下接口引用：CreateInstance, CreateLogstashInstance, DescribeInstances, DescribeLogstashInstances, UpdateInstance。

名称	类型	必选	描述
Zone	String	是	可用区 示例值：ap-beijing-1
SubnetId	String	是	子网ID 示例值：subnet-123abc
Hidden	Boolean	否	是否为隐藏可用区 示例值：false

错误码

最近更新时间：2025-06-26 01:20:41

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.UnAuthCreateInstance	当前用户对创建集群实例操作未授权，请添加CAM权限
AuthFailure.UnAuthDescribeInstances	当前用户对查询集群实例操作未授权，请添加CAM权限。
AuthFailure.UnAuthorizedOperation	操作未授权。
FailedOperation.AccessControlList	更新ACL失败
FailedOperation.CloseDataTier	关闭数据层失败，至少要保留热层和温层其中一层。
FailedOperation.ClusterResourceLimitError	集群资源配额限制错误。
FailedOperation.CosMigrate	cos融合迁移失败
FailedOperation.DiskCountParamError	节点磁盘块数参数检查失败。
FailedOperation.ErrorClusterState	集群状态错误。
FailedOperation.ErrorClusterStateUnhealth	集群状态不健康。
FailedOperation.EsDictionaryInfoError	由于EsDictionaryInfo错误，不允许操纵。
FailedOperation.FileNameError	文件名不符合规范。
FailedOperation.FileSizeError	文件大小不符合要求。
FailedOperation.GetTagInfoError	获取实例的标签列表错误。
FailedOperation.GetUser	获取user失败

错误码	说明
FailedOperation.GetUserCosSnapshotList	获取快照列表失败
FailedOperation.NoPayment	账户未绑定信用卡或paypal，无法支付。
FailedOperation.NodeNumAndZoneError	节点数和可用区的信息不符合对应操作的规则。
FailedOperation.NotAuthenticated	用户未实名认证。
FailedOperation.OpenDestroyProtection	当前集群已开启销毁保护，无法直接销毁，请解除该保护后再尝试销毁
FailedOperation.Password	解码Password时出错。
FailedOperation.RefundError	该实例不满足退费条件。
FailedOperation.RequestTimeOut	请求超时。
FailedOperation.SpaceLanguage	变更space的kibana语言信息失败
FailedOperation.TradeSignError	生成询价或者下单签名有误。
FailedOperation.UnsupportedLocalDiskRollUpScaleUpOrDown	不支持对本地盘集群通过滚动重启方式扩缩容。
FailedOperation.UnsupportedOpenLowFrequency	不支持开启低频存储
FailedOperation.UnsupportedResetNodeTypeAndScaleOutDisk	不支持在滚动重启扩容计算资源同时扩容磁盘数量。
FailedOperation.UnsupportedRestScaleDownAndModifyDisk	不支持在滚动重启缩容计算资源同时修改磁盘大小
FailedOperation.UnsupportedReverseRegulationNodeTypeAndDisk	不支持反向调节节点配置和磁盘容量。
FailedOperation.UpdateVpc	变更space的vpc信息失败
InvalidParameter.GC	Gc类型的取值和预期不符。
InvalidParameter.IndexNameExist	索引名已存在。
InvalidParameter.IndexNameInvalid	索引名不合法。
InvalidParameter.InvalidAppId	AppId的取值和预期不符。
InvalidParameter.InvalidAutoVoucher	是否自动使用代金券AutoVoucher的取值和预期不符。
InvalidParameter.InvalidBackends	Backends的取值和预期不符。
InvalidParameter.InvalidClusterName	ClusterName的取值和预期不符。
InvalidParameter.InvalidCosBackupInfo	cos自动备份信息与预期不符。
InvalidParameter.InvalidDeployMode	集群部署方式InvalidDeployMode的取值和预期不符。
InvalidParameter.InvalidDiskCount	节点磁盘块数DiskCount取值和预期不符。
InvalidParameter.InvalidDiskEncrypt	节点磁盘加密信息DiskEncrypt的取值与预期不符。
InvalidParameter.InvalidDiskEnhance	是否启用增强型ssd云盘DiskEnhance的取值和预期不符。
InvalidParameter.InvalidDiskSize	节点磁盘容量DiskSize的取值和预期不符。
InvalidParameter.InvalidDiskType	节点磁盘类型DiskType的取值和预期不符。
InvalidParameter.InvalidDns	内网域名解析服务器IP Dns的取值和预期不符。
InvalidParameter.InvalidEsACL	EsAcl的取值和预期不符。
InvalidParameter.InvalidEsVersion	实例版本EsVersion的取值和预期不符。
InvalidParameter.InvalidIndexMetaJson	创建的索引元数据JSON IndexMetaJson的取值和预期不符。
InvalidParameter.InvalidIndexName	索引名IndexName的取值和预期不符。
InvalidParameter.InvalidIndexOptionsField	索引自治字段IndexOptionsField的取值和预期不符。
InvalidParameter.InvalidIndexPolicyField	索引生命周期字段IndexPolicyField的取值和预期不符。

版权所有：腾讯云计算（北京）有限责任公司

错误码	说明
InvalidParameter.InvalidIndexSettingsField	索引配置字段IndexSettingsField的取值和预期不符。
InvalidParameter.InvalidIndexType	索引类型IndexType的取值和预期不符。
InvalidParameter.InvalidInstanceId	无效的InstanceId，没有找到对应资源。
InvalidParameter.InvalidInstanceOperationDurations	集群可维护时间段和集群ID InstanceOperationDurations的取值和预期不符。
InvalidParameter.InvalidIp	Ip地址取值和预期不符。
InvalidParameter.InvalidIpList	私有网络vip列表IpList的取值和预期不符。
InvalidParameter.InvalidKibanaPrivatePort	Kibana内网访问端口的取值和预期不符。
InvalidParameter.InvalidLimit	分页大小Limit的取值和预期不符。
InvalidParameter.InvalidLogType	日志类型LogType的取值和预期不符。
InvalidParameter.InvalidLogstashVersion	实例版本LogstashVersion的取值和预期不符。
InvalidParameter.InvalidMultiZoneInfo	多可用区部署ZoneDetail的信息与预期不符。
InvalidParameter.InvalidNodeNames	节点名称列表NodeNames的取值和预期不符。
InvalidParameter.InvalidNodeNum	节点数量NodeNum的取值和预期不符。
InvalidParameter.InvalidNodeType	节点规格NodeType的取值和预期不符。
InvalidParameter.InvalidNodeTypeStorageInfo	温热层数据结构NodeTypeInfos的取值和预期不符。
InvalidParameter.InvalidOffset	分页起始值Offset的取值和预期不符。
InvalidParameter.InvalidOpType	操作类型OpType的取值和预期不符。
InvalidParameter.InvalidOperationDuration	可维护时间段OperationDuration的取值和预期不符。
InvalidParameter.InvalidOrderByKey	排序字段OrderByKey的取值和预期不符。
InvalidParameter.InvalidOrderByType	排序方式OrderByType的取值和预期不符。
InvalidParameter.InvalidPara	需要签名的参数Para的取值和预期不符。
InvalidParameter.InvalidPrivateAccess	内网访问PrivateAccess的取值和预期不符。
InvalidParameter.InvalidPublicAccess	外网访问PublicAccess的取值和预期不符。
InvalidParameter.InvalidQueryString	无效的lucene语法
InvalidParameter.InvalidRegion	地域Region的取值与预期不符。
InvalidParameter.InvalidRestartMode	可选重启模式RestartMode的取值和预期不符。
InvalidParameter.InvalidRestartType	重启方式RestartType的取值和预期不符。
InvalidParameter.InvalidSampleJson	数据样例JSON SampleJson的取值和预期不符。
InvalidParameter.InvalidSecurityGroupIds	安全组id列表SecurityGroupIds的取值和预期不符。
InvalidParameter.InvalidServerlessId	ES serverless的serverlessId未找到
InvalidParameter.InvalidSpaceId	ES Serverless的spaceId未找到
InvalidParameter.InvalidSpaceNames	space的名称错误
InvalidParameter.InvalidSubnetId	子网ID SubnetId的取值与预期不符。
InvalidParameter.InvalidSubnetUidList	虚拟子网络统一ID列表的取值和预期不符。
InvalidParameter.InvalidTagInfo	节点标签信息TagInfo的取值和预期不符。
InvalidParameter.InvalidTagList	节点标签信息列表TagList的取值和预期不符。

错误码	说明
InvalidParameter.InvalidTargetIndexName	目标索引名TargetIndexName的取值和预期不符。
InvalidParameter.InvalidTargetNodeTypes	接收请求的目标节点类型列表InvalidTargetNodeTypes的取值和预期不符。
InvalidParameter.InvalidTimeParam	时间相关参数的取值或格式和预期不符。
InvalidParameter.InvalidTraceUuid	调用链Id TraceUuid的取值和预期不符。
InvalidParameter.InvalidType	节点类型Type的取值和预期不符。
InvalidParameter.InvalidUin	Uin的取值和预期不符。
InvalidParameter.InvalidUpdateMetaJson	更新的索引元数据json UpdateMetaJson的取值和预期不符。
InvalidParameter.InvalidUpdateType	更新类型UpdateType的取值和预期不符。
InvalidParameter.InvalidVoucherIds	代金券ID列表VoucherIds的取值和预期不符。
InvalidParameter.InvalidZone	可用区Zone的信息与预期不符。
InvalidParameter.SpaceNameExist	空间名已存在。
InvalidParameter.SpaceNameInvalid	空间名不合法。
InvalidParameter.UsernameExist	用户名已存在。
InvalidParameterValue.ChargePeriod	包年包月购买时长ChargePeriod的取值有误。
InvalidParameterValue.ChargeType	计费类型ChargeType的参数取值有误。
InvalidParameterValue.CheckOnly	是否只做升级检查CheckOnly的取值有误。
InvalidParameterValue.ConfigInfo	组件配置值有误。
InvalidParameterValue.EnableDoubleEni	是否允许双网卡EnableDoubleEni的值有误。
InvalidParameterValue.EsConfigType	ES配置类型取值有误。
InvalidParameterValue.EsPort	端口EsPort的取值有误。
InvalidParameterValue.EsVip	EsVip的取值有误。
InvalidParameterValue.InstallBundleList	要安装的bundle信息列表InstallBundleList的取值有误。
InvalidParameterValue.InstallPluginList	为集群稳定起见，当前只支持同时安装1个用户插件。
InvalidParameterValue.InstanceName	实例名校验失败，按规则输入。
InvalidParameterValue.InvalidDeadline	截止日期冲突。
InvalidParameterValue.InvalidJDK	Jdk类型的取值和预期不符。
InvalidParameterValue.Password	密码校验不合法。
InvalidParameterValue.PluginType	插件类型PluginType的取值有误。
InvalidParameterValue.RemoveBundleList	要卸载的bundle名列表RemoveBundleList的取值有误。
InvalidParameterValue.RemovePluginList	为集群稳定起见，当前只支持同时卸载1个用户插件。
InvalidParameterValue.RenewFlag	自动续费标识RenewFlag的参数取值有误。
InvalidParameterValue.TargetPath	要安装的bundle的zip包解压所在的路径TargetPath的取值有误。
InvalidParameterValue.TimeUnit	计费时长单位TimeUnit的取值有误。
InvalidParameterValue.UpgradeMode	升级方式UpgradeMode的取值有误。
LimitExceeded.ClusterNum	该账号下的集群数超过限额。
LimitExceeded.DiagnoseCount	超过当天智能诊断最大次数。
LimitExceeded.DiskCount	节点磁盘块数超过最大限额。

错误码	说明
LimitExceeded.IndexCount	空间索引数超过配额。
LimitExceeded.IndexCreate	空间索引每日创建数超过配额。
LimitExceeded.NodeNumOrIndices	节点数量或索引存储量超过限额。
LimitExceeded.PluginInstall	超过最大自定义插件数量，请先删除部分自定义插件。
LimitExceeded.ResourceLimit	超出一个用户可以拥有的最大资源限制。
LimitExceeded.SpaceCount	空间数超过配额。
LimitExceeded.SpaceCreate	空间每日创建数超过配额。
LimitExceeded.UpdateItemLimit	更新参数值过多，超过限制。
RegionError	地域错误
ResourceInUse.Diagnose	存在诊断中的作业，请等待作业诊断完成后重试。
ResourceInUse.IsolatedStatus	索引状态处于隔离状态，不能进行其他操作。
ResourceInUse.Order	订单被锁定。
ResourceInsufficient.Balance	账户余额不足。
ResourceInsufficient.CVM	计算资源不足，可更换可用区或调整机型。
ResourceInsufficient.CVMQuota	cvm份额不足。
ResourceInsufficient.Clb	clb资源不足
ResourceInsufficient.Es	es资源不足
ResourceInsufficient.HiddenZone	隐藏可用区专用主节点资源不足。
ResourceInsufficient.Subnet	子网剩余ip数量不足。
ResourceInsufficient.SubnetIp	可用的子网IP不足。
ResourceInsufficient.Zone	可用区资源不足。
ResourceNotFound.AccountInfoNotFound	账户信息未找到。
ResourceNotFound.BaradMetricNotFound	巴拉多（barad）指标资源未找到。
ResourceNotFound.CAMInfoNotFound	CAM资源未找到。
ResourceNotFound.CLBNotFound	负载均衡信息未找到。
ResourceNotFound.ClusterInfoNotFound	集群资源获取失败。
ResourceNotFound.CosInfoNotFound	COS相关信息获取失败。
ResourceNotFound.DBInfoNotFound	数据库资源获取失败。
ResourceNotFound.DataNodeNotFound	数据节点信息未找到。
ResourceNotFound.DiagnoseNotFound	智能诊断相关资源未找到。
ResourceNotFound.DiskInfoNotFound	磁盘相关资源获取失败。
ResourceNotFound.EsInfoNotFound	集群信息获取失败
ResourceNotFound.KMSNotFound	密钥管理系统资源获取失败。
ResourceNotFound.OssInfoNotFound	获取oss资源失败。
ResourceNotFound.STSNotFound	获取安全凭证服务资源失败。
ResourceNotFound.SecurityGroupNotFound	安全组信息获取失败。
ResourceNotFound.TradeCgwNotFound	获取计费资源失败。

版权所有：腾讯云计算（北京）有限责任公司

错误码	说明
ResourceNotFound.VPCInfoNotFound	VPC资源获取失败。
ResourceNotFound.WhiteListNotFound	白名单资源获取失败。
UnauthorizedOperation.UinNotInWhiteList	Uin不在白名单中。
UnsupportedOperation.AuditStatusConflict	审计日志的状态冲突。
UnsupportedOperation.BasicSecurityType	不支持该认证方式。
UnsupportedOperation.ChangeNodeType	不支持变配节点或磁盘的配置。
UnsupportedOperation.ClusterShardNum	不支持该操作，在集群中存在分配的分片。
UnsupportedOperation.ClusterStateClose	集群中存在部分索引处于关闭状态。
UnsupportedOperation.ClusterStateNoReplication	集群中存在部分索引无备份。
UnsupportedOperation.ClusterStateUnHealth	集群的状态不正常（不为绿）。
UnsupportedOperation.CosBackUp	该版本不支持cos备份。集群需要重启来允许cos备份。
UnsupportedOperation.CvmStatusStopped	有关机的cvm
UnsupportedOperation.DiagnoseJob	不支持该智能诊断任务。
UnsupportedOperation.DiagnoseNotOpen	智能诊断未开启。
UnsupportedOperation.DiskFreeSpaceInsufficient	节点磁盘可用空间不足
UnsupportedOperation.DiskUse	不支持的读硬盘配置。
UnsupportedOperation.EditListLength	不支持这个长度的editList的操作。
UnsupportedOperation.HeteroScalingExecuting	存在未完成的异构扩缩容流程
UnsupportedOperation.IndexSettingsRequireSet	集群有索引设置index.routing.allocation.require._name,or _ip, or _id
UnsupportedOperation.InstanceTypeError	不支持该操作，服务类型有误。
UnsupportedOperation.LicenseError	不支持该操作，license的类型有误。
UnsupportedOperation.LocalDisk	不支持本地盘机型。
UnsupportedOperation.Migrate	不支持迁移操作。
UnsupportedOperation.MultiZonesUpgrade	不支持多可用区变配。
UnsupportedOperation.PassLogstashId	用户同步日志的配置passLogstashId为空，不支持该操作。
UnsupportedOperation.Plugin	不支持该插件。
UnsupportedOperation.Protocol	不支持对应Protocol的操作。
UnsupportedOperation.ReadRate	不支持的读取速度。
UnsupportedOperation.RestartMode	离线节点蓝绿变更存在风险，请直接选择重启“in-place”。
UnsupportedOperation.ScaleDownTooMuch	纵向缩容太多资源。
UnsupportedOperation.StatusNotNormal	不支持该操作，当前实例运行状态不为正常。
UnsupportedOperation.StatusNotSupport	不支持该操作，实例状态有误。
UnsupportedOperation.UpdateDiskEncrypt	节点或磁盘数变更时不支持更新磁盘加密信息。
UnsupportedOperation.UpdateDiskType	不支持更新磁盘类型。
UnsupportedOperation.UpdateNodeNumAndNodeType	不支持同时更新节点个数（NodeNum）和节点规格（NodeType）。
UnsupportedOperation.VPCInfoNotFound	VPC资源获取失败。
UnsupportedOperation.WebServiceType	不支持的web组件。

错误码	说明
UnsupportedOperation.WriteRate	不支持的写入速度。

Elasticsearch Service API 2025-01-01
产品版本

最近更新时间：2025-06-20 01:34:33

您正在浏览es产品文档的版本: 2025-01-01

最新版本

- 2025-01-01 (当前版本)

以往版本

- 2018-04-16

更新历史

最近更新时间：2025-07-09 01:25:18

第 6 次发布

发布时间：2025-07-09 01:25:15

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- Document
  - 新增成员：FileStartPageNumber, FileEndPageNumber

第 5 次发布

发布时间：2025-06-23 01:34:33

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- ChunkDocument
  - 新增出参：Usage

修改数据结构：

- ChunkConfig
  - 新增成员：ChunkOverlap

第 4 次发布

发布时间：2025-06-20 01:34:30

本次发布包含了以下内容：

改善已有的文档。

新增数据结构：

- DocumentParseConfig

修改数据结构：

- ParseDocument
  - 新增成员：DocumentParseConfig, FileStartPageNumber, FileEndPageNumber

第 3 次发布

发布时间：2025-06-04 01:34:18

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- WebSearch

删除数据结构：

- WebPage

第 2 次发布

发布时间：2025-04-18 01:13:14

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- WebSearch

新增数据结构：

- WebPage

第 1 次发布

发布时间：2025-04-10 16:47:28

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ChatCompletions
- ChunkDocument
- ChunkDocumentAsync
- GetDocumentChunkResult
- GetDocumentParseResult
- GetTextEmbedding
- ParseDocument
- ParseDocumentAsync
- RunRerank

新增数据结构：

- Choice
- Chunk
- ChunkConfig
- ChunkConfigAsync
- ChunkDocument
- Document
- DocumentChunkUsage
- EmbeddingData
- Message
- OnlineSearchOptions
- OutputMessage
- ParseDocument
- RerankResult
- TokenUsage
- Usage
- WebContent

简介

最近更新时间：2025-04-25 01:32:04

腾讯云 Elasticsearch Service（ES）是腾讯云基于开源搜索引擎 Elasticsearch 打造的高可用、可伸缩的云端托管 Elasticsearch 服务，部署在 VPC 内，集成了 Elasticsearch、Kibana 等组件。使用腾讯云 ES ，您可以快速部署、轻松管理、按需扩展您的 Elastic 集群，简化集群部署、管理等复杂运维操作，快速构建日志分析、异常监控、网站搜索导航、企业级搜索、BI 分析等各类业务。

API 概览

最近更新时间：2025-06-13 01:33:27

智能搜索开发相关接口

接口名称	接口功能	频率限制（次/秒）
ParseDocument	实时文档解析	20
ParseDocumentAsync	文档解析（异步）	20
GetDocumentParseResult	获取文档解析结果	20
ChunkDocument	在线文本切片	20
ChunkDocumentAsync	文档切片（异步）	20
GetDocumentChunkResult	获取文档切片结果	20
GetTextEmbedding	获取特征向量	1000
RunRerank	重排序	1000
ChatCompletions	大模型生成	200

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 01:19:06

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 es.tencentcloudapi.com ，也支持指定地域域名访问，例如广州地域的域名为 es.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 es.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）	es.tencentcloudapi.com
华南地区（广州）	es.ap-guangzhou.tencentcloudapi.com
华东地区（上海）	es.ap-shanghai.tencentcloudapi.com
华东地区（南京）	es.ap-nanjing.tencentcloudapi.com
华北地区（北京）	es.ap-beijing.tencentcloudapi.com
西南地区（成都）	es.ap-chengdu.tencentcloudapi.com
西南地区（重庆）	es.ap-chongqing.tencentcloudapi.com
港澳台地区（中国香港）	es.ap-hongkong.tencentcloudapi.com
亚太东南（新加坡）	es.ap-singapore.tencentcloudapi.com
亚太东南（雅加达）	es.ap-jakarta.tencentcloudapi.com
亚太东南（曼谷）	es.ap-bangkok.tencentcloudapi.com
亚太东北（首尔）	es.ap-seoul.tencentcloudapi.com
亚太东北（东京）	es.ap-tokyo.tencentcloudapi.com
美国东部（弗吉尼亚）	es.na-ashburn.tencentcloudapi.com
美国西部（硅谷）	es.na-siliconvalley.tencentcloudapi.com
南美地区（圣保罗）	es.sa-saopaulo.tencentcloudapi.com
欧洲地区（法兰克福）	es.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）	es.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	es.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：2025-04-10 16:47:41

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 es；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
华北地区（北京）	ap-beijing

签名方法 v3

最近更新时间：2025-04-10 16:47:41

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）。

版权所有：腾讯云计算（北京）有限责任公司

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：2025-04-10 16:47:41

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：2025-04-10 16:47:42

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：2025-04-10 16:47:42

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。

智能搜索开发相关接口
实时文档解析

最近更新时间：2025-06-05 21:35:59

1. 接口描述

接口请求域名： es.ai.tencentcloudapi.com 。

本服务可将各类格式文档精准转换为标准格式，满足企业知识库建设、技术文档迁移、内容平台结构化存储等需求。
本接口有单账号调用上限控制，如您有提高并发限制的需求请联系我们。

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：ParseDocument。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
Document	是	ParseDocument	文件信息 示例值：{ "FileType": "TXT", "FileUrl": "", "FileContent": "data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==", "FileName": "test.txt" }
ModelName	是	String	模型名称，doc-llm。 示例值：doc-llm

3. 输出参数

参数名称	类型	描述
Progress	String	进度：0-100。 示例值：100
DocumentParseResultUrl	String	解析文件结果。 示例值：https://ai-search-doc-bj-xxxxx.cos.ap-beijing.myqcloud.com/&2F1257780094/2013/81e11a5472dd4288a0b998c032b4df26.zip
FailedPages	Array of Integer	失败页码。 示例值：[1]
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。本接口为流式响应接口，当请求成功时，RequestId 会被放在 HTTP 响应的 Header "X-TC-RequestId" 中。

4. 示例

示例1 文档解析示例

输入示例

```
POST / HTTP/1.1
Host: es.tencentcloudapi.com
```

```
Content-Type: application/json
X-TC-Action: ParseDocument
<公共请求参数>

{
"Document": {
"FileType": "TXT",
"FileUrl": "",
"FileContent": "data:text/plain;base64,SGVsbG8sIFdvcmxkIQ=="
},
"ModelName": "doc-llm"
}
```

输出示例

```
{
"Response": {
"RequestId": "7ab3fd45-61e7-4e5e-8194-b1997f67fc18",
"Progress": "0",
"DocumentParseResultUrl": "",
"FailedPages": []
}
}
```

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

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	参数取值错误。
MissingParameter	缺少参数错误。

错误码	描述
UnknownParameter	未知参数错误。

文档解析（异步）

最近更新时间：2025-05-08 01:33:33

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

本服务可将各类格式文档精准转换为标准格式，满足企业知识库建设、技术文档迁移、内容平台结构化存储等需求。
本接口为异步接口，有单账号调用上限控制，如您有提高并发限制的需求请联系我们 。

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：ParseDocumentAsync。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
Document	是	Document	文件信息。 示例值：{ "FileType": "TXT", "FileUrl": "", "FileContent": "data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==", "FileName": "test.txt" }
ModelName	是	String	模型名称，可选模型列表：doc-llm。 示例值：doc-llm

3. 输出参数

参数名称	类型	描述
TaskId	String	任务 id 示例值：6538a10a-18c5-4cba-a870-b02e78b46f71
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 文件解析异步任务

输入示例

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

{
"Document": {
"FileType": "TXT",
"FileUrl": "",
"FileContent": "data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==",
"FileName": "test.txt"
},
"ModelName": "doc-llm"
}
```

输出示例

```
{
"Response": {
"RequestId": "e3a1a515-e53c-473f-9114-6d05a19d1c0c",
"TaskId": "acf20b7e-47b2-4cfd-8aba-d170dee34cfc"
}
}
```

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。
LimitExceeded	超过配额限制。

获取文档解析结果

最近更新时间：2025-07-01 01:22:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

本接口用于获取文档解析异步处理结果。

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：GetDocumentParseResult。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
TaskId	是	String	任务 Id 示例值：10343124

3. 输出参数

参数名称	类型	描述
Status	Integer	任务状态，-1：失败，0：运行中，1：成功。 示例值：1
DocumentParseResultUrl	String	结果文件。 示例值：https://ai-search-doc-bj-xxxxx.cos.ap-beijing.myqcloud.com/&2F1257780094/2013/81e11a5472dd4288a0b998c032b4df26.zip
FailedPages	Array of Integer	失败的页码。 注意：此字段可能返回 null，表示取不到有效值。 示例值：1,2
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 获取文档解析任务结果

输入示例

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

{
```

```
"TaskId": "3cce395a-fe9e-460f-9b19-8920d15867e3"
}
```

输出示例

```
{
"Response": {
"DocumentParseResultUrl": "https://aisearch-xxxx.cos.ap-xxx.myqcloud.com/%2F1257780094/2025/87a8419e7e7d4bd9ad17267c47d121
68.zip?x-cos-security-token=xxx&q-sign-algorithm=sha1&q-ak=xxx&q-sign-time=1744014678%3B1744101078&q-key-time=1744014678%3
B1744101078&q-header-list=host&q-url-param-list=x-cos-security-token&q-signature=xxx",
"FailedPages": null,
"RequestId": "7da936b2-8d10-42bd-b65f-8d10ae99927a",
"Status": 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
InternalError	内部错误。
InvalidParameter	参数错误。
LimitExceeded	超过配额限制。

在线文本切片

最近更新时间：2025-06-23 01:34:35

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

文本切片是将长文本分割为短片段的技术，用于适配模型输入、提升处理效率或信息检索，平衡片段长度与语义连贯性，适用于NLP、数据分析等场景。
本接口为分隔符规则切片接口，有单账号调用上限控制，如您有提高并发限制的需求请 联系我们 。

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：ChunkDocument。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
Document	是	ChunkDocument	切片文件信息。 示例值：Document
ModelName	是	String	模型名称，可选模型列表：doc-chunk。 示例值：doc-chunk
Config	否	ChunkConfig	文件切片配置。 示例值：Config

3. 输出参数

参数名称	类型	描述
Chunks	Array of Chunk	无
Usage	Usage	token消耗量
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 实时文档切片

输入示例

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

{
```

```
"Document": {
"FileType": "TXT",
"FileContent": "基于分隔符、文本长度进行切片，适用规则性较强的文本"
},
"ModelName": "doc-chunk",
"Config": {
"MaxChunkSize": 100
}
}
```

输出示例

```
{
"Response": {
"Chunks": [
{
"Content": "基于分隔符、文本长度进行切片，适用规则性较强的文本",
"Index": 0
}
],
"RequestId": "eeb0d39a-8a4a-431b-a577-2595202afaa1"
}
}
```

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。

文档切片（异步）

最近更新时间：2025-05-08 01:33:35

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

文本切片是将长文本分割为短片段的技术，用于适配模型输入、提升处理效率或信息检索，平衡片段长度与语义连贯性，适用于NLP、数据分析等场景。
本接口为异步接口，有单账号调用上限控制，如您有提高并发限制的需求请联系我们 。

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：ChunkDocumentAsync。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
Document	是	Document	文件信息。 示例值：{"FileType":"TXT"}
ModelName	是	String	模型名称，可选模型列表：doc-tree-chunk。 示例值：doc-tree-chunk
Config	否	ChunkConfigAsync	文件切片配置。 示例值：{"MaxChunkSize":"20"}

3. 输出参数

参数名称	类型	描述
TaskId	String	任务 ID 示例值：acf20b7e-47b2-4cfd-8aba-d170dee34cfc
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 文档切片异步

输入示例

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

{
"Document": {
```

```
"FileType": "TXT",
"FileUrl": "",
"FileContent": "data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==",
"FileName": "test.txt"
},
"Config": {
"MaxChunkSize": 10000
},
"ModelName": "doc-tree-chunk"
}
```

输出示例

```
{
"Response": {
"RequestId": "e3a1a515-e53c-473f-9114-6d05a19d1c0c",
"TaskId": "acf20b7e-47b2-4cfd-8aba-d170dee34cfc"
}
}
```

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。
LimitExceeded	超过配额限制。

获取文档切片结果

最近更新时间：2025-07-01 01:22:48

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

获取文档切片结果

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：GetDocumentChunkResult。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
TaskId	是	String	任务 ID 示例值：3041d1dc-1072-4025-a3c2-f8baea25801b

3. 输出参数

参数名称	类型	描述
Status	Integer	任务状态，-1：失败，0：运行中，1：成功。 示例值：1
DocumentChunkResultUrl	String	切片结果文件。 示例值：https://ai-search-doc-bj-xxxxx.cos.ap-beijing.myqcloud.com/&2F1257780094/2013/81e11a5472dd4288a0b998c032b4df26.zip
Usage	DocumentChunkUsage	Token用量。 示例值：{"TotalTokens":12}
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 获取文档切片结果

输入示例

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

{
```

```
"TaskId": "3041d1dc-1072-4025-a3c2-f8baea25801b"
}
```

输出示例

```
{
"Response": {
"DocumentChunkResultUrl": "https://aisearch-xxx.cos.ap-xxx.myqcloud.com/%2F1257780094/2032/0eef71d5c6b94f4c9f77bf96a990dd93.zip?x-cos-security-token=xxx=sha1&q-ak=AKID_xxx&q-sign-time=1744101492%3B1744187892&q-key-time=1744101492%3B1744187892&q-header-list=host&q-url-param-list=x-cos-security-token&q-signature=xxx",
"RequestId": "0b2bf3af-5671-45e8-a5db-xxx",
"Status": 1,
"Usage": {
"PageNumber": 1,
"TotalTokens": 118
}
}
}
```

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。
LimitExceeded	超过配额限制。

获取特征向量

最近更新时间：2025-06-13 01:33:26

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

Embedding是一种将高维数据映射到低维空间的技术，通常用于将非结构化数据，如文本、图像或音频转化为向量表示，使其更容易输入机器模型进行处理，并且向量之间的距离可以反映对象之间的相似性。
本接口有单账号调用上限控制，如您有提高并发限制的需求请联系我们 。

默认接口请求频率限制：1000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：GetTextEmbedding。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
ModelName	是	String	模型名称，可选模型列表：bge-base-zh-v1.5,conan-embedding-v1,bge-m3,KaLM-embedding-multilingual-mini-v1。 示例值：bge-base-zh-v1.5
Texts.N	是	Array of String	需进行向量化的文本集。 示例值：["今天天气怎样"]

3. 输出参数

参数名称	类型	描述
Data	Array of EmbeddingData	结果集
Usage	Usage	向量化消耗的token数量。
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 获取特征向量

输入示例

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

{
```

```
"ModelName": "bge-large-zh-v1.5",
"Texts": [
"你好"
]
}
```

输出示例

```
{
"Response": {
"Data": [
{
"Embedding": [
-0.007741480600088835,
-0.017921222373843193,
-0.04486401379108429,
-0.0073757413774728775,
0.014934351667761803,
-0.025723660364747047,
0.03559861704707146,
-0.034379489719867706,
0.027674268931150436,
0.04876523092389107,
-0.04266957566142082
],
"Index": 0
}
],
"RequestId": "8750abd0-feab-4249-ba96-2df9a14c7202",
"Usage": {
"TotalTokens": 2
}
}
}
```

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	参数取值错误。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceNotFound	资源不存在。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

版权所有：腾讯云计算（北京）有限责任公司

重排序

最近更新时间：2025-06-13 01:33:26

1. 接口描述

接口请求域名： es.tencentcloudapi.com 。

重排是指在 RAG 过程中，通过评估文档与查询之间的相关性，将最相关的文档放在前面，确保语言模型在生成回答时优先考虑排名靠前的上下文，提高生成结果的准确性和可信度，也可以通过这种方式进行过滤，减少大模型成本。
本接口有单账号调用上限控制，如您有提高并发限制的需求请联系我们 。

默认接口请求频率限制：1000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：RunRerank。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
ModelName	是	String	模型名称，可选模型列表：bge-reranker-large，bge-reranker-v2-m3。 示例值：bge-reranker-large
Query	是	String	查询文本。 示例值：今天天气怎样
Documents.N	是	Array of String	待排序的候选doc列表。 示例值：["今天是晴天"]
TopN	否	Integer	排序返回的top文档数量, 如果没有指定则返回全部候选doc，如果指定的top_n值大于输入的候选doc数量，返回全部doc。 示例值：5
ReturnDocuments	否	Boolean	返回的排序结果列表里面是否返回每一条document原文，默认值False。 示例值：true

3. 输出参数

参数名称	类型	描述
Data	Array of RerankResult	输出结果集。
Usage	Usage	消耗token数量。 示例值：224
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 重排序

输入示例

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

{
"ModelName": "bge-rerank-large",
"Query": "你好",
"Documents": [
"你好？",
"你好。"
]
}
```

输出示例

```
{
"Response": {
"Data": [
{
"Document": "",
"Index": 1,
"RelevanceScore": 0.9998511075973511
},
{
"Document": "",
"Index": 0,
"RelevanceScore": 0.9996849298477173
}
],
"RequestId": "cf9c5598-9625-436d-bc81-8dd3900e63c0",
"Usage": {
"TotalTokens": 10
}
}
}
```

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

版权所有：腾讯云计算（北京）有限责任公司

- 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	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
RequestLimitExceeded	请求的次数超过了频率限制。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。

大模型生成

最近更新时间：2025-06-05 21:35:59

1. 接口描述

接口请求域名： es.ai.tencentcloudapi.com 。

本服务支持一系列高性能的大语言模型，包括DeepSeek以及腾讯自主研发的混元大模型，结合混合搜索等先进搜索技术，快速高效实现RAG，有效解决幻觉和知识更新问题。
本接口有单账号调用上限控制，如您有提高并发限制的需求请联系我们 。

默认接口请求频率限制：200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：ChatCompletions。
Version	是	String	公共参数，本接口取值：2025-01-01。
Region	是	String	公共参数，详见产品支持的 地域列表。
Messages.N	是	Array of Message	会话内容，按对话时间从旧到新在数组中排列，长度受模型窗口大小限制。 示例值：[{}]
ModelName	是	String	模型名称，可选模型列表：hunyuan-turbo，hunyuan-large，hunyuan-large-longcontext，hunyuan-standard，hunyuan-standard-256K，deepseek-r1，deepseek-v3，deepseek-r1-distill-qwen-32b。 示例值：hunyuan-turbo
Stream	否	Boolean	是否以流式接口的形式返回数据，默认true。 示例值：是否以流式接口的形式返回数据，默认false
TopP	否	Float	取值区间为[0.0, 1.0], 非必要不建议使用, 不合理的取值会影响效果 。 示例值：0.1
Temperature	否	Float	取值区间为[0.0, 2.0], 非必要不建议使用, 不合理的取值会影响效果 。 示例值：1.1
OnlineSearch	否	Boolean	是否开启联网搜索。默认为 false。 示例值：false
OnlineSearchOptions	否	OnlineSearchOptions	当 OnlineSearch 为 true 时，指定的搜索引擎，默认为 bing。 示例值：{"Engine": "bing"}

3. 输出参数

参数名称	类型	描述
Id	String	此次请求的id 示例值：8f71dbaf64194da7ac3b61b145aa3534
Choices	Array of Choice	回复内容
Usage	TokenUsage	token使用量 示例值：{"PromptTokens":324,"CompletionTokens":309,"TotalTokens":633}

参数名称	类型	描述
OnlineSearchContent	Array of WebContent	联网搜索结果。 注意：此字段可能返回 null，表示取不到有效值。 示例值：[{}]
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。本接口为流式响应接口，当请求成功时，RequestId 会被放在 HTTP 响应的 Header "X-TC-RequestId" 中。

4. 示例

示例1 AI 聊天问答

输入示例

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

{
"Messages": [
{
"Role": "user",
"Content": "什么是大模型？ "
}
],
"ModelName": "deepseek-r1",
"Stream": false,
"OnlineSearch": true,
"OnlineSearchOptions": {
"Engine": "bing"
}
}
```

输出示例

```
{
"Response": {
"Id": "59afbba4-8b5f-47fa-972d-4bc5d77004ee",
"Usage": {
"PromptTokens": 9224,
"CompletionTokens": 203,
"TotalTokens": 9427
},
"Choices": [
{
"Message": {
"Role": "assistant",
"Content": "\n\n大模型是参数量庞大、基于深度学习的AI模型，擅长处理复杂任务如自然语言处理。",
"ReasoningContent": "\n嗯，用户让我简单描述什么是大模型，我需要先理解大模型的核心定义。根据搜索结果，大模型通常指参数规模庞大、基于深度学习的AI模型，比如GPT系列。多个网页提到参数数量达到数十亿甚至万亿，处理复杂任务能力强。例如，腾讯云和今日头条的文章都强调大模型的参数和数据处理能力。\n"
}
}
],
"RequestId": "6b5a7f7a-3e3d-4974-8480-8e21edaec5e7",
```

```
"OnlineSearchContent": [
{
"Query": "大模型 解释",
"Title": "快速了解什么是大模型 - 今日头条",
"Url": "https://m.toutiao.com/video/7340170115901932086/?upstream_biz=toutiao_pc",
"Time": "2024-02-27",
"Content": "大模型(Large Model)是AI人工智能领域中的一种重要模型,通常指的是参数量非常大、数据量也非常大的深度学习模型.大模型通常由数百万到数十亿的参数组成,需要大量的数据和计算资源进行训练和推理.",
"ChunkIndex": "ChunkIndex",
"Score": "Score"
}
]
}
}
```


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

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	参数取值错误。
MissingParameter	缺少参数错误。
UnknownParameter	未知参数错误。

数据结构

最近更新时间：2025-07-09 01:25:17

Choice

返回的回复, 支持多个。

被如下接口引用：ChatCompletions。

名称	类型	必选	描述
Message	OutputMessage	否	返回的回复。

Chunk

切片对象信息

被如下接口引用：ChunkDocument。

名称	类型	必选	描述
Index	Integer	否	chunk索引。切片顺序 id。 示例值：0
Content	String	否	chunk内容。 示例值：今天天气很好

ChunkConfig

文档分片配置

被如下接口引用：ChunkDocument。

名称	类型	必选	描述
MaxChunkSize	Integer	否	最大分片长度 示例值：4800
Delimiters	Array of String	否	分隔符列表 示例值：["\n\n", "\n", "。", "！ ", "？ ", "， ", ""]
ChunkOverlap	Integer	否	相邻切片重合字符数，需要小于分片长度 示例值：4000

ChunkConfigAsync

文档切片异步任务

被如下接口引用：ChunkDocumentAsync。

名称	类型	必选	描述
MaxChunkSize	Integer	否	最大分片长度 示例值：4800

ChunkDocument

切片文档信息

被如下接口引用：ChunkDocument。

名称	类型	必选	描述
FileType	String	否	文件类型，支持 MD，TXT 格式。 示例值：MD

名称	类型	必选	描述
FileContent	String	否	文本原文，使用字符串格式输入。 示例值：53847824487347829

Document

文档信息

被如下接口引用：ChunkDocumentAsync, ParseDocumentAsync。

名称	类型	必选	描述
FileType	String	是	文件类型。 支持的文件类型：PDF、DOC、DOCX、PPT、PPTX、MD、TXT、XLS、XLSX、CSV、PNG、JPG、JPEG、BMP、GIF、WEBP、HEIC、EPS、ICNS、IM、PCX、PPM、TIFF、XBM、HEIF、JP2 支持的文件大小： - PDF、DOC、DOCX、PPT、PPTX 支持100M - MD、TXT、XLS、XLSX、CSV 支持10M - 其他支持20M 示例值：md
FileUrl	String	否	文件存储于腾讯云的 URL 可保障更高的下载速度和稳定性，使用腾讯云COS 文件地址。 示例值：https://example.com/example.pdf
FileContent	String	否	文件的 base64 值，携带 MineType前缀信息。编码后的后的文件不超过 10M。 支持的文件大小：所下载文件经Base64编码后不超过 8M。文件下载时间不超过3秒。 支持的图片像素：单边介于20-10000px之间。 示例值：data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==
FileName	String	否	文件名称，当使用 base64上传的时候使用。 示例值：example.txt
FileStartPageNumber	Integer	否	文档的起始页码 示例值：1
FileEndPageNumber	Integer	否	文档的结束页码 示例值：1

DocumentChunkUsage

文档切片用量

被如下接口引用：GetDocumentChunkResult。

名称	类型	必选	描述
PageNumber	Integer	否	解析页面数量 示例值：1
TotalTokens	Integer	否	消耗 token数量 示例值：1

DocumentParseConfig

文档解析配置

被如下接口引用：ParseDocument。

名称	类型	必选	描述
ImageResponseType	Integer	否	0:图片以链接形式返回 1:返回图片中提取的文本内容 示例值：1

EmbeddingData

向量内容

被如下接口引用：GetTextEmbedding。

名称	类型	描述
Embedding	Array of Float	embedding 内容 注意：此字段可能返回 null，表示取不到有效值。 示例值：[3.234,13.4556]
Index	Integer	索引序号 注意：此字段可能返回 null，表示取不到有效值。 示例值：3

Message

会话内容，按对话时间从旧到新在数组中排列，长度受模型窗口大小限制。

被如下接口引用：ChatCompletions。

名称	类型	必选	描述
Role	String	否	角色，‘system'，‘user'，'assistant'或者'tool', 在message中, 除了system，其他必须是user与assistant交替(一问一答) 示例值：user
Content	String	否	具体文本内容 示例值：请问如何访问腾讯云官网?

OnlineSearchOptions

联网搜索选项。

被如下接口引用：ChatCompletions。

名称	类型	必选	描述
Engine	String	否	搜索引擎。支持 bing 和 sogou。 示例值：bing

OutputMessage

会话内容，按对话时间从旧到新在数组中排列，长度受模型窗口大小限制。

被如下接口引用：ChatCompletions。

名称	类型	必选	描述
Role	String	否	角色 示例值：user
Content	String	否	文本内容 示例值：深圳今天的天气是晴转多云。
ReasoningContent	String	否	推理内容 示例值：用户询问天气，需要联网搜索结果。

ParseDocument

文档信息

被如下接口引用：ParseDocument。

名称	类型	必选	描述
FileType	String	是	文件类型。 支持的文件类型：PDF、DOC、DOCX、PPT、PPTX、MD、TXT、XLS、XLSX、CSV、PNG、JPG、JPEG、BMP、GIF、WEBP、HEIC、EPS、ICNS、IM、PCX、PPM、TIFF、XBM、HEIF、JP2

版权所有：腾讯云计算（北京）有限责任公司

名称	类型	必选	描述
			支持的文件大小： - PDF、DOC、DOCX、PPT、PPTX 支持100M - MD、TXT、XLS、XLSX、CSV 支持10M - 其他支持20M 示例值：md
FileUrl	String	否	文件存储于腾讯云的 URL 可保障更高的下载速度和稳定性，使用腾讯云COS 文件地址。 示例值：https://example.com/example.pdf
FileContent	String	否	文件的 base64 值，携带 MineType前缀信息。编码后的后的文件不超过 10M。 支持的文件大小：所下载文件经Base64编码后不超过 8M。文件下载时间不超过3秒。 支持的图片像素：单边介于20-10000px之间。 示例值：data:text/plain;base64,SGVsbG8sIFdvcmxkIQ==
DocumentParseConfig	DocumentParseConfig	否	文档解析配置
FileStartPageNumber	Integer	否	文档的起始页码 示例值：1
FileEndPageNumber	Integer	否	文档的结束页码 示例值：1

RerankResult

输出结果

被如下接口引用：RunRerank。

名称	类型	描述
Index	Integer	对应的doc在输入候选doc数组中的位置索引值 注意：此字段可能返回 null，表示取不到有效值。 示例值：4
RelevanceScore	Float	相似度分数 注意：此字段可能返回 null，表示取不到有效值。 示例值：43.2
Document	String	doc原文内容 注意：此字段可能返回 null，表示取不到有效值。 示例值：今天天气怎样

TokenUsage

token使用量

被如下接口引用：ChatCompletions。

名称	类型	必选	描述
PromptTokens	Integer	否	表示prompt的tokens数，多次返回中保持不变 示例值：324
CompletionTokens	Integer	否	回答的token总数，在流式返回中，表示到目前为止所有completion的tokens总数，多次返回中持续累加 示例值：309
TotalTokens	Integer	否	表示prompt_tokens和completion_tokens之和 示例值：633

Usage

token消耗总数

被如下接口引用：ChunkDocument, GetTextEmbedding, RunRerank。

名称	类型	描述
TotalTokens	Integer	tokens总数 注意：此字段可能返回 null，表示取不到有效值。 示例值：32

WebContent

搜索结果网页信息。

被如下接口引用：ChatCompletions。

名称	类型	描述
Query	String	搜索问题 示例值：今天 深圳 天气 2025
Title	String	标题 示例值：标题
Url	String	链接 示例值：https://example.com
Time	String	时间 示例值：16小时前
Content	String	网页内容 示例值：深圳龙岗，今日小雨。
ChunkIndex	String	切片索引 示例值：0
Score	String	分数 示例值：0

版权所有：腾讯云计算（北京）有限责任公司

错误码

最近更新时间：2025-04-10 16:47:42

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。