

前端性能监控 API 文档



腾讯云

【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】



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

控制台相关接口

创建发布文件记录

创建白名单

删除发布文件

删除白名单

获取QueryData

获取dataloginfo信息

获取PvUrlInfo信息

获取首页错误信息

获取PV列表

获取应用对应sourcemap文件列表

查询片区信息

获取UV列表

获取白名单列表

获取存储临时密钥

APP查询相关接口

查询 app 多维分析数据

获取 app 监控指标数据

查询 app 监控个例样本详情列表

查询 app 监控个例聚合列表

日志相关接口

获取Rum日志聚合信息

导出Rum日志列表

获取Rum日志导出列表

获取Rum日志列表

获取Rum分钟级日志列表

业务系统和项目信息相关接口

创建 RUM 应用

添加星标应用

删除 RUM 业务系统

删除 RUM 应用

删除星标应用

获取上报量

获取应用上报抽样信息

获取 RUM 应用列表

获取首页分数列表

查询 RUM 业务系统信息

修改 RUM 业务系统

修改 RUM 应用信息

新增修改限流

恢复 RUM 业务系统

恢复 RUM 应用

停止RUM业务系统

停止应用上报

创建 RUM 业务系统

页面性能相关接口

获取PerformancePage信息

获取DescribeDataSetUrlStatistics信息

获取DescribeDataWebVitalsPage信息

自定义测速相关接口

获取DescribeDataCustomUrl信息

自定义事件相关接口

获取DescribeDataEventUrl信息

API监控相关接口

获取DescribeDataFetchProject信息（已下线，请使用DescribeDataFetchUrl）

获取DescribeDataFetchUrl信息

获取DescribeDataFetchUrlInfo信息

异常分析相关接口

获取LogUrlStatistics信息

页面访问相关接口

获取DescribeDataPvUrlStatistics信息

静态资源监控相关接口

获取DescribeDataStaticProject信息

获取DescribeDataStaticResource信息

获取DescribeDataStaticUrl信息

数据结构

错误码

API 文档

更新历史

最近更新时间：2025-05-27 01:54:44

第 43 次发布

发布时间：2025-05-27 01:54:31

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataPerformancePage](#)
 - 新增入参：WebVitals

第 42 次发布

发布时间：2025-03-07 01:24:12

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [ModifyProject](#)
 - 新增入参：EnableKafka, KafkaHost, KafkaTopic, KafkaVersion, SaslUserName, SaslPassword, SaslMechanism, SinkId
- [ResumeInstance](#)
 - 新增入参：IsModifyAll
- [StopInstance](#)
 - 新增入参：IsModifyAll

第 41 次发布

发布时间：2025-02-14 01:54:47

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeTawInstances](#)
 - 废弃入参：InstanceStatuses

第 40 次发布

发布时间：2025-02-12 01:54:12

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeProjects](#)
 - 废弃入参：IsDemo
- [DescribeScores](#)
 - 废弃入参：IsDemo

- [DescribeTawInstances](#)
 - 废弃入参: InstanceIds, IsDemo

第 39 次发布

发布时间: 2025-01-22 01:23:51

本次发布包含了以下内容:

改善已有的文档。

新增数据结构:

- [Kafka](#)

修改数据结构:

- [ReleaseFile](#)
 - 修改成员: Version, FileKey, FileName, FileHash, ID
- [RumInstanceInfo](#)
 - 修改成员: InstanceType
- [RumProject](#)
 - 新增成员: Kafka
 - 修改成员: AccessPoint

第 38 次发布

发布时间: 2025-01-14 01:23:14

本次发布包含了以下内容:

改善已有的文档。

修改数据结构:

- [ScoreInfo](#)
 - 新增成员: PagePerformanceScore, JsErrorScore, ApiPerformanceScore, ApiAvaliableScore, StaticPerformanceScore, StaticAvaliableScore
 - 修改成员: CreateTime

第 37 次发布

发布时间: 2024-11-18 01:52:29

本次发布包含了以下内容:

改善已有的文档。

修改接口:

- [DescribeReleaseFileSign](#)
 - 新增入参: ID

第 36 次发布

发布时间: 2024-08-23 02:05:36

本次发布包含了以下内容:

改善已有的文档。

修改接口:

- [DescribeDataLogUrlStatistics](#)
 - 新增入参: ErrorMsg

第 35 次发布

发布时间: 2024-07-12 01:22:59

本次发布包含了以下内容:

改善已有的文档。

修改接口:

- [DescribeReleaseFiles](#)
 - 新增入参: FileName

第 34 次发布

发布时间: 2024-07-10 01:22:53

本次发布包含了以下内容:

改善已有的文档。

修改接口:

- [DescribeReleaseFileSign](#)
 - 新增入参: Site

第 33 次发布

发布时间: 2024-01-31 01:20:58

本次发布包含了以下内容:

改善已有的文档。

删除接口:

- DescribeDataPerformanceProject

第 32 次发布

发布时间: 2024-01-23 01:26:09

本次发布包含了以下内容:

改善已有的文档。

删除接口:

- CreateLogExport
- DeleteLogExport
- DescribeLogExports
- DescribeLogList

删除数据结构:

- LogExport

第 31 次发布

发布时间: 2024-01-12 01:21:17

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DeleteOfflineLogConfig
- DeleteOfflineLogRecord
- DescribeOfflineLogConfigs

第 30 次发布

发布时间：2024-01-10 01:21:10

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- CreateOfflineLogConfig
- DescribeOfflineLogRecords
- DescribeOfflineLogs

第 29 次发布

发布时间：2023-10-13 01:22:31

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataPvUrlStatistics](#)
 - 新增入参：IsNewData

第 28 次发布

发布时间：2023-08-09 01:43:06

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataPvUrlStatistics](#)
 - 新增入参：GroupByType

第 27 次发布

发布时间：2023-05-12 01:43:18

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [RumProject](#)
 - 新增成员：AccessPoint

第 26 次发布

发布时间：2023-04-26 01:43:23

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateTawInstance](#)
 - 新增入参：InstanceType

修改数据结构：

- [RumInstanceInfo](#)
 - 新增成员：InstanceType

第 25 次发布

发布时间：2023-04-18 01:36:49

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeAppDimensionMetrics](#)
- [DescribeAppMetricsData](#)
- [DescribeAppSingleCaseDetailList](#)
- [DescribeAppSingleCaseList](#)

第 24 次发布

发布时间：2023-04-10 01:49:56

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeReleaseFileSign](#)
 - 新增入参：FileType

第 23 次发布

发布时间：2023-02-10 01:54:01

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeRumLogExport](#)
- [DescribeRumLogExports](#)
- [ResumeProject](#)

第 22 次发布

发布时间：2023-01-12 01:32:36

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [ScoreInfo](#)
 - 新增成员：CreateTime

第 21 次发布

发布时间：2023-01-05 01:37:14

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeRunGroupLog](#)

第 20 次发布

发布时间：2022-12-23 01:49:31

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeRunLogList](#)
- [DescribeRunStatsLogList](#)

第 19 次发布

发布时间：2022-12-22 01:56:30

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [StopProject](#)

第 18 次发布

发布时间：2022-12-09 01:53:36

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateTawInstance](#)
 - 新增入参：ResourcePackageType, ResourcePackageNum
 - 新增出参：DealName

第 17 次发布

发布时间：2022-11-22 06:49:05

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateWhitelist](#)
 - 新增出参：ID

第 16 次发布

发布时间：2022-10-10 06:36:43

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataFetchUrl](#)
 - 新增入参：NetStatus
- [DescribeDataPerformancePage](#)
 - 新增入参：NetStatus

第 15 次发布

发布时间：2022-09-09 06:38:13

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- [RumAreaInfo](#)
 - 新增成员：AreaRegionID, AreaRegionCode, AreaAbbr

第 14 次发布

发布时间：2022-07-13 06:15:23

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataFetchProject](#)
 - 新增入参：Status, Ret
- [DescribeDataFetchUrl](#)
 - 新增入参：Status, Ret
- [DescribeDataSetUrlStatistics](#)
 - 新增入参：PackageType

第 13 次发布

发布时间：2022-03-21 08:15:09

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeProjects](#)
 - 新增入参：IsDemo
- [DescribeScores](#)

- 新增入参：IsDemo
- [DescribeTawInstances](#)
 - 新增入参：IsDemo

修改数据结构：

- [ProjectLimit](#)
 - 修改成员：ID, ProjectID
- [RunProject](#)
 - 新增成员：ProjectStatus

第 12 次发布

发布时间：2022-03-15 08:13:58

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [CreateTawInstance](#)
 - 新增入参：BuyingChannel

第 11 次发布

发布时间：2022-02-14 08:15:11

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateLogExport
- CreateOfflineLogConfig
- [CreateReleaseFile](#)
- [CreateStarProject](#)
- [CreateTawInstance](#)
- [CreateWhitelist](#)
- [DeleteInstance](#)
- DeleteLogExport
- DeleteOfflineLogConfig
- DeleteOfflineLogRecord
- [DeleteProject](#)
- [DeleteReleaseFile](#)
- [DeleteStarProject](#)
- [DeleteWhitelist](#)
- [DescribeData](#)
- [DescribeDataCustomUrl](#)
- [DescribeDataFetchProject](#)
- [DescribeDataLogUrlInfo](#)
- DescribeDataPerformanceProject
- [DescribeDataPvUrlInfo](#)
- [DescribeDataReportCount](#)
- [DescribeDataSetUrlStatistics](#)
- [DescribeDataStaticProject](#)
- [DescribeDataStaticResource](#)

- [DescribeDataStaticUrl](#)
- [DescribeDataWebVitalsPage](#)
- [DescribeLogExports](#)
- [DescribeOfflineLogConfigs](#)
- [DescribeOfflineLogRecords](#)
- [DescribeOfflineLogs](#)
- [DescribeProjectLimits](#)
- [DescribePvList](#)
- [DescribeReleaseFileSign](#)
- [DescribeReleaseFiles](#)
- [DescribeTawAreas](#)
- [DescribeTawInstances](#)
- [DescribeUvList](#)
- [DescribeWhitelists](#)
- [ModifyInstance](#)
- [ModifyProject](#)
- [ModifyProjectLimit](#)
- [ResumeInstance](#)
- [StopInstance](#)

新增数据结构：

- [LogExport](#)
- [ProjectLimit](#)
- [ReleaseFile](#)
- [RumAreaInfo](#)
- [RumInstanceInfo](#)
- [RumPvInfo](#)
- [RumUvInfo](#)
- [Tag](#)
- [Whitelist](#)

第 10 次发布

发布时间：2022-01-18 08:14:02

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeDataFetchUrl](#)
- [DescribeDataFetchUrlInfo](#)

第 9 次发布

发布时间：2021-12-14 08:13:36

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- [DescribeDataEventUrl](#)
 - 新增入参：Env
- [DescribeDataLogUrlStatistics](#)

- 新增入参：Env
- [DescribeDataPerformancePage](#)
 - 新增入参：Env
- [DescribeDataPvUrlStatistics](#)
 - 新增入参：Env

第 8 次发布

发布时间：2021-12-09 08:09:35

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeDataEventUrl](#)
- [DescribeDataPvUrlStatistics](#)

第 7 次发布

发布时间：2021-09-27 08:15:19

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeScores](#)

新增数据结构：

- [ScoreInfo](#)

第 6 次发布

发布时间：2021-09-14 08:00:07

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeProjects](#)

新增数据结构：

- [Filter](#)
- [RumProject](#)

第 5 次发布

发布时间：2021-09-13 08:00:06

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeLogList](#)

第 4 次发布

发布时间：2021-08-20 08:00:08

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeDataLogUrlStatistics](#)

第 3 次发布

发布时间：2021-08-19 08:00:11

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeError](#)

第 2 次发布

发布时间：2021-08-09 08:00:05

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [DescribeDataPerformancePage](#)

第 1 次发布

发布时间：2021-08-04 08:00:04

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [CreateProject](#)

简介

最近更新时间：2025-04-25 01:51:38

前端性能监控（Real User Monitoring，RUM）是一站式前端监控解决方案，专注于 Web 和小程序等大前端领域，主要关注用户页面性能（页面测速、接口测速、CDN 测速等）和质量（JS 错误、Ajax 错误等），并且联动腾讯云应用性能监控实现前后端一体化监控。用户只需要安装 SDK 到自己的项目中，通过简单配置化，即可实现对用户页面质量的全方位守护，真正做到了低成本使用和无侵入监控。

用户可以使用本文档介绍的 API 对应用监控服务进行相关操作，如读取监控数据等，具体支持的操作可参见 [API 概览](#)。

注意：

- 本章节云监控 API 接口均为最新 API 3.0 接口，后续相关新增功能都会在此章节更新。我们强烈推荐您使用最新 API3.0 接口。

API 概览

最近更新时间：2024-08-02 02:01:39

控制台相关接口

接口名称	接口功能	频率限制（次/秒）
CreateReleaseFile	创建发布文件记录	20
CreateWhitelist	创建白名单	20
DeleteReleaseFile	删除发布文件	20
DeleteWhitelist	删除白名单	20
DescribeData	获取QueryData	20
DescribeDataLogUrlInfo	获取dataloginfo信息	20
DescribeDataPvUrlInfo	获取PvUrlInfo信息	20
DescribeError	获取首页错误信息	20
DescribePvList	获取PV列表	20
DescribeReleaseFiles	获取应用对应sourcemap文件列表	20
DescribeTawAreas	查询片区信息	20
DescribeUvList	获取UV列表	20
DescribeWhitelists	获取白名单列表	20
DescribeReleaseFileSign	获取存储临时密钥	20

APP查询相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeAppDimensionMetrics	查询 app 多维分析数据	20
DescribeAppMetricsData	获取 app 监控指标数据	50
DescribeAppSingleCaseDetailList	查询 app 监控个例样本详情列表	20
DescribeAppSingleCaseList	查询 app 监控个例聚合列表	20

日志相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeRumGroupLog	获取Rum日志聚合信息	20
DescribeRumLogExport	导出Rum日志列表	20
DescribeRumLogExports	获取Rum日志导出列表	20
DescribeRumLogList	获取Rum日志列表	20
DescribeRumStatsLogList	获取Rum分钟级日志列表	20

业务系统和项目信息相关接口

接口名称	接口功能	频率限制（次/秒）
CreateProject	创建 RUM 应用	20
CreateStarProject	添加星标应用	20
DeleteInstance	删除 RUM 业务系统	20
DeleteProject	删除 RUM 应用	20
DeleteStarProject	删除星标应用	20
DescribeDataReportCount	获取上报量	20
DescribeProjectLimits	获取应用上报抽样信息	20
DescribeProjects	获取 RUM 应用列表	20
DescribeScores	获取首页分数列表	20
DescribeTawlInstances	查询 RUM 业务系统信息	20
ModifyInstance	修改 RUM 业务系统	20
ModifyProject	修改 RUM 应用信息	20
ModifyProjectLimit	新增修改限流	20
ResumeInstance	恢复 RUM 业务系统	20
ResumeProject	恢复 RUM 应用	20
StopInstance	停止RUM业务系统	20
StopProject	停止应用上报	20
CreateTawlInstance	创建 RUM 业务系统	20

页面性能相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataPerformancePage	获取PerformancePage信息	20
DescribeDataSetUrlStatistics	获取DescribeDataSetUrlStatistics信息	20
DescribeDataWebVitalsPage	获取DescribeDataWebVitalsPage信息	20

自定义测速相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataCustomUrl	获取DescribeDataCustomUrl信息	20

自定义事件相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataEventUrl	获取DescribeDataEventUrl信息	20

API监控相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataFetchProject	获取DescribeDataFetchProject信息（已下线，请使用DescribeDataFetchUrl）	20
DescribeDataFetchUrl	获取DescribeDataFetchUrl信息	20
DescribeDataFetchUrlInfo	获取DescribeDataFetchUrlInfo信息	20

异常分析相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataLogUrlStatistics	获取LogUrlStatistics信息	20

页面访问相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataPvUrlStatistics	获取DescribeDataPvUrlStatistics信息	20

静态资源监控相关接口

接口名称	接口功能	频率限制（次/秒）
DescribeDataStaticProject	获取DescribeDataStaticProject信息	20
DescribeDataStaticResource	获取DescribeDataStaticResource信息	20
DescribeDataStaticUrl	获取DescribeDataStaticUrl信息	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 01:28:24

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

注意：由于金融区和非金融区是隔离不通的，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）	rum.ap-shanghai-fsi.tencentcloudapi.com
华南地区（深圳金融）	rum.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：2024-12-10 01:53:43

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 rum；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

参数名称	类型	必选	描述
			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 。
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
华南地区（广州）	ap-guangzhou

地域	取值
亚太东南（新加坡）	ap-singapore
欧洲地区（法兰克福）	eu-frankfurt

签名方法 v3

最近更新时间：2024-12-25 01:55:53

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 中间号（？）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。

字段名称	解释
	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")
```

衍生出的密钥 SecretDate、SecretService 和 SecretSigning 是二进制的数，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;host;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=content-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-na\nme\"}]}";
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: '.$saction.'"
.' -H "X-TC-Timestamp: '.$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=#{signature}"
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++)
    {
        sprintf(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);

#ifdef 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++)
    {
        sprintf(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 01:55:54

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：

- 登录 [腾讯云管理中心控制台](#)。
- 前往 [云 API 密钥](#) 的控制台页面
- 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
InstanceId.0	待查询的实例 ID	ins-09dx96dg

参数名称	中文	参数值
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"，因此格式化后即为 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。

签名原串的连接规则为：请求方法 + 请求主机 + 请求路径 + ? + 请求字符串。

示例的连接结果为：


```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&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&Region=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 TENCLCLOUD_SECRET_ID, 值为示例的 AKID*****
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCLCLOUD_SECRET_KEY, 值为示例的 *****
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$params["Nonce"] = 11886;//rand();
$params["Timestamp"] = 1465185768;//time();
$params["Region"] = "ap-guangzhou";
$params["SecretId"] = $secretId;
$params["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 requestHost, 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 01:45:54

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 06:44:50

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。

控制台相关接口

创建发布文件记录

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

创建对应项目的文件记录

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：CreateReleaseFile。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	项目 id 示例值：1
Files.N	是	Array of ReleaseFile	文件信息列表

3. 输出参数

参数名称	类型	描述
Msg	String	调用结果 示例值：success
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 创建文件记录

输入示例

```
https://rum.tencentcloudapi.com/?Action=CreateReleaseFile
&ProjectID=1
&Files.0.Version='0.0.1'
&Files.0.FileName='a.js.map'
&Files.0.FileKey='1-0.0.1-a.js.map'
```

```
&Files.0.FileHash='XNdLOW5RN***'
```

&<公共请求参数>

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalError	内部错误。
InvalidParameter	参数错误。

错误码	描述
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

创建白名单

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

创建白名单

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：CreateWhitelist。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	实例ID：taw-123 示例值：taw-123
Remark	是	String	备注（暂未作字节数限制） 示例值：备注
WhitelistUin	是	String	uin：业务方标识 示例值：136
Aid	否	String	业务方标识 示例值：2222

3. 输出参数

参数名称	类型	描述
Msg	String	消息 示例值：success
ID	Integer	白名单ID 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 12

输入示例


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

```
{
  "InstanceId": "taw-123",
  "Aid": "333",
  "Remark": "222",
  "WhitelistUin": "22323"
}
```

输出示例

```
{
  "Response": {
    "ID": 77,
    "Msg": "success",
    "RequestId": "fcaec326-8c68-4302-8624-c5b08b9f1893"
  }
}
```

示例2 创建白名单

输入示例

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

```
{
  "InstanceId": "taw-123",
  "Aid": "222",
  "Remark": "备注",
  "WhitelistUin": "1123"
}
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "ID": 1,
    "RequestId": "d61a85dc-d9d6-431e-812d-095eb4e129db"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。

错误码	描述
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除发布文件

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

将对应 sourcemap 文件删除

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：DeleteReleaseFile。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ID	是	Integer	文件 id 示例值：1

3. 输出参数

参数名称	类型	描述
Msg	String	接口请求返回字符串 示例值：success
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 删除文件记录

输入示例

```
https://rum.tencentcloudapi.com/?Action=DeleteReleaseFile
&ID=1
&<公共请求参数>
```

输出示例

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

```
"Msg": "success",
"RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除白名单

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

删除白名单

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：DeleteWhitelist。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	实例ID 示例值：taw-123
ID	是	String	名单ID 示例值：1

3. 输出参数

参数名称	类型	描述
Msg	String	消息success 示例值：success
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 删除白名单

输入示例

```
https://rum.tencentcloudapi.com/?Action=DeleteWhitelist
&ID=1
&InstanceId=taw-123
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalServerError	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。
OperationDenied	操作被拒绝。

错误码	描述
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取QueryData

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

转发monitor查询

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：DescribeData。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Query	是	String	查询字符串 示例值: "select delta(count) as allCount from db.name where time >= 1621240040s and time <= 1621243620s and id='1613' group by "ext3" limit 1000"
ID	是	Integer	项目ID 示例值: 1

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值: "{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]}",
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 转发获取data信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeData
&ID=1
&Query="select delta(count) as allCount from db.name where time >= 1621240040s and time <= 1621243620s and id='1
```

```
613' group by \"ext3\" limit 1000"
```

&<公共请求参数>

输出示例

```
{
  "Response": {
    "Result": "{\"request_id\":\"65a8fec7-2b39-4b11-893f-3715279d235f\",\"results\": [{\"statement_id\":0,\"total\":0},{\"statement_id\":1,\"total\":0}]}\",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalError	内部错误。

错误码	描述
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取dataloginfo信息

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取loginfo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：DescribeDataLogUrlInfo。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ID	是	Integer	项目ID 示例值：1
StartTime	是	Integer	时间戳 示例值：1625444040
EndTime	是	Integer	时间戳 示例值：1625454840

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]}
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取dataloginfo信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataLogUrlInfo
&ID=1
&StartTime=1625444040
```

&EndTime=1625454840

&<公共请求参数>

输出示例

```
{
  "Response": {
    "Result": "{\\\"request_id\\\":\\\"65a8fec7-2b39-4b11-893f-3715279d235f\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0}]}\",
    \"RequestId\": \"65a8fec7-2b39-4b11-893f-3715279d235f\"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalError	内部错误。

错误码	描述
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取PvUrlInfo信息

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取PvUrlInfo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：DescribeDataPvUrlInfo。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	类型 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
lsp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国

参数名称	必选	类型	描述
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]}"
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取PvUrlInfo信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataPvUrlInfo
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
```

```
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Result": "{ \"request_id\": \"65a8fec7-2b39-4b11-893f-3715279d235f\", \"results\": [{ \"statement_id\": 0, \"total\": 0 }, { \"statement_id\": 1, \"total\": 0 } ] }",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。

错误码	描述
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取首页错误信息

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取首页错误信息

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：DescribeError。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Date	是	String	日期 示例值：20210520
ID	是	Integer	项目ID 示例值：1

3. 输出参数

参数名称	类型	描述
Content	String	内容 示例值："request error"
ID	Integer	项目ID 示例值：1
CreateTime	String	时间 示例值：20210520
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 获取首页错误信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeError
&ID=1
```

```
&Date="20210520"
&<公共请求参数>
{
  "Date": "20210520",
  "ID": 1
}
```

输出示例

```
{
  "Response": {
    "ID": 1,
    "CreateTime": "20210520",
    "Content": "request error",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败

错误码	描述
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取PV列表

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取项目下的PV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：DescribePvList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectId	是	Integer	ID 示例值：1
EndTime	是	String	结束时间 示例值：20201104
StartTime	是	String	开始时间 示例值：20201101
Dimension	否	String	对PV指标的查询维度。获取day：d， 获取min则不填。 示例值：d

3. 输出参数

参数名称	类型	描述
ProjectPvSet	Array of RumPvInfo	pv列表 示例值： [{"ProjectId":26,"Pv":236,"CreateTime":20241123}]
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PV列表

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribePvList
&ProjectId=1
```

```
&StartTime="1"
&EndTime="20"
&Dimension="d"
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "ProjectPvSet": [
      {
        "ProjectId": 26,
        "Pv": "236",
        "CreateTime": "20241123"
      }
    ],
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。

错误码	描述
FailedOperation	操作失败。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取应用对应sourcemap文件列表

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取应用对应sourcemap文件列表

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：DescribeReleaseFiles。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	项目 id 示例值：1
FileVersion	否	String	文件版本 示例值：0.0.1
FileName	否	String	查询过滤条件（根据sourcemap的文件名模糊匹配） 示例值：test.js

3. 输出参数

参数名称	类型	描述
Files	Array of ReleaseFile	文件信息列表 示例值：[{"Version":"0.01","FileKey":"test","FileName":"a.js.map","FileHash":"test"}]
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 获取应用对应sourcemap文件列表

获取应用对应sourcemap文件列表

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeReleaseFiles
&ProjectID=1
&<公共请求参数>
```

```
{
  "ProjectID": 2,
  "FileName": "2"
}
```

输出示例

```
{
  "Response": {
    "Files": [
      {
        "Version": "0.01",
        "FileKey": "*****test",
        "FileName": "a.js.map",
        "FileHash": "*****test"
      }
    ],
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。

错误码	描述
FailedOperation	操作失败。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

查询片区信息

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

查询片区信息

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：DescribeTawAreas。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Arealds.N	否	Array of Integer	片区Id 示例值：1
AreaKeys.N	否	Array of String	片区Key 示例值：area-cn-dl
Limit	否	Integer	分页Limit，默认根据AreaKeys和AreaStatuses参数查询所有。 示例值：1
AreaStatuses.N	否	Array of Integer	片区状态(1=有效，2=无效) 示例值：1
Offset	否	Integer	分页Offset，默认根据AreaKeys和AreaStatuses参数查询所有。 示例值：100

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	片区总数 示例值：1
AreaSet	Array of RumAreaInfo	片区列表
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 查询片区

输入示例

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

{
  "AreaIds": [
    1
  ],
  "AreaStatuses": [
    1
  ],
  "Limit": 0,
  "Offset": 100
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 1,
    "AreaSet": [
      {
        "AreaId": 1,
        "AreaStatus": 1,
        "AreaName": "中国大陆",
        "AreaKey": "area_cn_d1"
      }
    ],
    "RequestId": "10f605a3-6480-4254-bbb7-7ca1ab6a133e"
  }
}
```

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。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。

获取UV列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取项目下的UV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：DescribeUvList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectId	是	Integer	ID 示例值：1
EndTime	是	String	结束时间 示例值：20201104
StartTime	是	String	开始时间 示例值：20201101
Dimension	否	String	获取day: d, min:m 示例值：d

3. 输出参数

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

4. 示例

示例1 获取UV列表

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeUvList
&ProjectId=1
```



```
&StartTime="1"
&EndTime="20"
&Dimension="d"
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "ProjectUvSet": [],
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalError	内部错误。

错误码	描述
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取白名单列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取白名单列表

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：DescribeWhitelists。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	实例instance-ID 示例值：taw-123

3. 输出参数

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

4. 示例

示例1 获取白名单列表

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeWhitelists
&InstanceId=taw-123
&<公共请求参数>
```

输出示例

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

```
"WhitelistSet": [],
"RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
}
```

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
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取存储临时密钥

最近更新时间：2025-04-25 01:51:34

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取上传文件存储的临时密钥

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：DescribeReleaseFileSign。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Timeout	否	Integer	超时时间，不填默认是 5 分钟 示例值：10
FileType	否	Integer	bucket类型，不填默认1:web, 2:app 示例值：2
Site	否	Integer	获取临时签名的bucket是国内站还是国际站（1表示国际站，其它表示国内站） 示例值：0
ID	否	Integer	ProjectID 示例值：12000

3. 输出参数

参数名称	类型	描述
SecretKey	String	临时密钥key 示例值：OnKjAEea4W5***
SecretID	String	临时密钥 id 示例值：AKIDDHDvz_o***
SessionToken	String	临时密钥临时 token 示例值：2nTxEweEOsBo***
StartTime	Integer	开始时间戳 示例值：1739258276
ExpiredTime	Integer	过期时间戳 示例值：1739260076

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 获取临时密钥

获取临时密钥

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeReleaseFileSign
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "SecretKey": "OnKjAEa4W5***",
    "SecretID": "AKIDDHDvz_cW***",
    "SessionToken": "2nTxEweEOsBoGaz0sjm***",
    "StartTime": 1739258276,
    "ExpiredTime": 1739260076,
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

APP查询相关接口

查询 app 多维分析数据

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

用于查询 app 监控多维分析数据

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：DescribeAppDimensionMetrics。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	app 项目ID 示例值：1
From	是	String	查询的表名 示例值：kv_7_dist
Fields	是	String	查询指标 fields 示例值：[["as"],["if"],["equals","launch_cnt"],["toInt64",0]],0,[
Filter	是	String	查询的过滤条件 示例值：["or"],["and"],["equals","category","PERF_CRASH"],["i
FilterSimple	否	String	查询简单过滤条件 示例值：{"category__in":["PERF_CRASH","APP_LAUNCH","PERF_MASTER_T
GroupBy.N	否	Array of String	group by 条件 示例值：["request_cnt"]
OrderBy.N	否	Array of String	order by 条件 示例值：["request_cnt desc"]
Limit	否	Integer	limit 参数 示例值：1
Offset	否	Integer	offset 参数 示例值：1
BusinessContext	否	String	业务上下文参数 示例值：{"threshold":"0.001"}

3. 输出参数

参数名称	类型	描述
Data	String	查询数据返回 示例值: {\n "status": "ok",\n "code": 10000,\n "msg": \
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 查询多维分析数据

输入示例

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

{
  "Filter": "[]'",
  "ProjectID": 1,
  "FilterSimple": "'{\\\\"entrance_time__gte\\":1671367462000,\\\\"entrance_time__lte\\":1671453862000,\\\\"category_in\\":[\\\\"PERF_NET_REQUEST\\"],\\\\"d4\\":\\\\"false\\",\\\\"d3\\":\\\\"http\\",\\\\"v1__lte\\":300000}'",
  "From": "'kv_7_dist'",
  "Fields": "'[\\\\"explodeDimensions\\",\\\\"d2\\",\\\\"d8\\",[\\\\"as\\",[\\\\"multiIf\\",[\\\\"equals\\",\\\\"dimension_name\\",\\\\"d2'\\"],\\\\"domain'\\",[\\\\"equals\\",\\\\"dimension_name\\",\\\\"d8'\\"],\\\\"region'\\",\\\\"dimension_name\\",\\\\"dimension\\",[\\\\"as\\",[\\\\"avg\\",\\\\"v1\\"],\\\\"cost_time\\",[\\\\"as\\",[\\\\"count\\",1],\\\\"request_cnt\\",[\\\\"as\\",[\\\\"aggByPartition\\",\\\\"sum\\",\\\\"request_cnt\\",\\\\"dimension_name\\",\\\\"total_request_cnt\\",[\\\\"as\\",\\\\"total_request_cnt\\",\\\\"total_cnt\\",[\\\\"as\\",\\\\"request_cnt\\",\\\\"cnt\\",[\\\\"as\\",[\\\\"aggByPartition\\",\\\\"row_number\\",\\\\"none\\",\\\\"dimension_name\\",\\\\"order_by\\",\\\\"request_cnt\\",\\\\"desc\\"]],\\\\"rank_cnt\\"]]'",
  "Limit": 1000
}
```

输出示例

```
{
  "Response": {
    "Data": "'{\\\\"status\\":\\\\"ok\\",\\\\"code\\":10000,\\\\"msg\\":\\\\"ok\\",\\\\"data\\":[{\\\\"by_dimension\\":[\\\\"d2\\",\\\\"m.zipkin.com\\"],\\\\"cnt\\":1280,\\\\"cost_time\\":1480,\\\\"dimension\\":\\\\"domain\\",\\\\"dimension_name\\":\\\\"d2\\",\\\\"dimension_value\\":\\\\"m.zipkin.com\\",\\\\"rank_cnt\\":1,\\\\"request_cnt\\":1280,\\\\"total_cnt\\":1280,\\\\"total_request_cnt\\":1280},{\\\\"by_dimension\\":[\\\\"d8\\",\\\\"unsupported\\"],\\\\"cnt\\":1280,\\\\"cost_time\\":1480,\\\\"dimension\\":\\\\"region\\",\\\\"dimension_name\\":\\\\"d8\\",\\\\"dimension_value\\":\\\\"unsupported\\",\\\\"rank_cnt\\":1,\\\\"request_cnt\\":1280,\\\\"total_cnt\\":1280,\\\\"total_request_cnt\\":1280}]}'",
    "RequestId": "8a211fc4-7ce2-40ce-ad49-1c29cca2f3fc"
  }
}
```

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。
ResourceNotFound	资源不存在。

获取 app 监控指标数据

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取 app 监控指标数据

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：DescribeAppMetricsData。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	app 项目ID 示例值：1
From	是	String	查询的表名 示例值：kv_7_dist
Fields	是	String	查询指标 field 示例值：[["as"],["if"],["equals","launch_cnt"],["toInt64",0]],0,[
Filter	是	String	查询的过滤条件 示例值：["or"],["and"],["equals","category","PERF_CRASH"],["i
FilterSimple	否	String	查询简单过滤条件 示例值：{"category__in":["PERF_CRASH","APP_LAUNCH","PERF_MASTER_T
GroupBy.N	否	Array of String	group by 条件 示例值：["launch_cnt"]
OrderBy.N	否	Array of String	order by 条件 示例值：["launch_cnt desc"]
Limit	否	Integer	limit 参数 示例值：1
Offset	否	Integer	offset 参数 示例值：1
GroupByModifier	否	String	group by 参数 示例值：rollup

3. 输出参数

输出示例

```
{
  "Response": {
    "Data": "{\n  \"status\": \"ok\",\n  \"code\": 10000,\n  \"msg\": \"ok\",\n  \"data\": [\n    {\n      \"crash_cnt\": 18782,\n      \"crash_rate\": 0.7724132258595163,\n      \"crash_user_cnt\": 1,\n      \"crash_user_rate\": 1,\n      \"launch_cnt\": 24316,\n      \"total_cnt\": 24316,\n      \"user_cnt\": 1\n    }\n  ]\n}",
    "RequestId": "77b45639-2b70-40b5-b9af-739488787109"
  }
}
```

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 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。
ResourceNotFound	资源不存在。

查询 app 监控个例样本详情列表

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

查询 app 监控个例样本详情列表

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：DescribeAppSingleCaseDetailList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	app 项目ID 示例值：1
From	是	String	查询的表名 示例值：singleton_v2_dist
Fields	是	String	查询指标 field 示例值：["[\"as\",\"device_id\",\"device_id\"],\"entrance_time\", [\"as\",\"feature_md5\",\"feature_md5\"],[\"as\",\"version\",\"version\"],[\"as\",\"id\",\"id\"], [\"as\",\"user_id\",\"user_id\"]"]
Filter	是	String	查询的过滤条件 示例值：{"category__in":["PERF_CRASH"]}
FilterSimple	否	String	查询简单过滤条件 示例值：{"category__in":["PERF_CRASH","APP_LAUNCH","PERF_MASTER_T
GroupBy.N	否	Array of String	group by 条件 示例值：["entrance_time"]
OrderBy.N	否	Array of String	order by 条件 示例值：["entrance_time desc"]
Limit	否	Integer	limit 参数 示例值：8
Offset	否	Integer	offset 参数 示例值：1

3. 输出参数

参数名称	类型	描述
Data	String	查询数据返回 示例值: {\n "status": "ok",\n "code": 10000,\n "msg": \
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 app 监控个例详情列表

输入示例

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

{
  "Filter": "{\n \"category__in\": [\"PERF_CRASH\"] }",
  "ProjectID": 1,
  "FilterSimple": "{\n \"entrance_time__gte\": 1671025353000, \"entrance_time__lte\": 1671284553000, \"feature_md5\": \"8a
ba3b8538d7d843c558012c11219e1d\", \"category__in\": [\"PERF_CRASH\"] }",
  "From": "db.name",
  "Fields": "[[\"as\", \"device_id\", \"device_id\"], \"entrance_time\", [\"as\", \"feature_md5\", \"feature_md5\"], [\"a
s\", \"version\", \"version\"], [\"as\", \"id\", \"id\"], [\"as\", \"user_id\", \"user_id\"]]",
  "Limit": 200,
  "OrderBy": [
    "entrance_time desc"
  ]
}
```

输出示例

```
{
  "Response": {
    "Data": "{\n \"status\": \"ok\", \n \"code\": 10000, \n \"msg\": \"ok\", \n \"data\": [\n {\n \"crash_cnt\": 1878
2, \n \"crash_rate\": 0.7724132258595163, \n \"crash_user_cnt\": 1, \n \"crash_user_rate\": 1, \n \"launch_cnt\": 24
316, \n \"total_cnt\": 24316, \n \"user_cnt\": 1\n } \n ] \n }",
    "RequestId": "77b45639-2b70-40b5-b9af-739488787109"
  }
}
```

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 app 监控个例聚合列表

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

查询 app 监控个例聚合列表

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：DescribeAppSingleCaseList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	app 项目 ID 示例值：1
From	是	String	查询的表名 示例值：kv_7_dist
Fields	是	String	查询指标 field 示例值：[["as",["if",["equals","launch_cnt",["toInt64",0]],0,[
Filter	是	String	查询的过滤条件 示例值：["or",["and",["equals","category","PERF_CRASH"],["i
FilterSimple	否	String	查询简单过滤条件 示例值：{"category__in":["PERF_CRASH","APP_LAUNCH","PERF_MASTER_T
GroupBy.N	否	Array of String	group by 条件 示例值：["app_id"]
OrderBy.N	否	Array of String	order by 条件 示例值：["crash_cnt desc"]
Limit	否	Integer	limit 参数 示例值：1
Offset	否	Integer	offset 参数 示例值：1

3. 输出参数

参数名称	类型	描述
Data	String	查询数据返回 示例值: {\n "status": "ok",\n "code": 10000,\n "msg": \
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 app 个例聚合列表

输入示例

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

{
  "Filter": "{\n"app_id\n": 1}",
  "ProjectID": 1,
  "FilterSimple": "{\n"entrance_time__gte\n":1671370185000,\n"entrance_time__lte\n":1671456585000,\n"category__in\n":
[\n"PERF_CRASH\n"]}",
  "From": "'singleton_v2_dist'",
  "Fields": "[\n"app_id\n",[\n"as\n",[\n"any\n",[\n"cast\n",\n"feature\n",\n"'String'\n"]],\n"row_feature\n",[\n"as\n",[\n"max
\n",\n"entrance_time\n",\n"last_entrance_time\n",\n"feature_md5\n",[\n"as\n",[\n"uniq\n",\n"device_id\n",\n"crash_user_cnt
\n",[\n"as\n",[\n"count\n",1],\n"crash_cnt\n",[\n"as\n",[\n"aggByPartition\n",\n"sum\n",\n"crash_user_cnt\n",\n"app_id\n",\n"to
tal_crash_user_cnt\n",[\n"as\n",[\n"aggByPartition\n",\n"sum\n",\n"crash_cnt\n",\n"app_id\n",\n"total_crash_cnt\n"]]",
  "GroupBy": [
    "feature_md5",
    "app_id",
    "platform"
  ],
  "OrderBy": [
    "crash_cnt desc"
  ]
}
```

输出示例

```
{
  "Response": {
    "Data": "'{\n"status\n":\n"ok\n",\n"code\n":10000,\n"msg\n":\n"ok\n",\n"data\n":[]}'",
    "RequestId": "a2bcd13d-000f-4b84-a72e-6d4229079014"
  }
}
```

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。
ResourceNotFound	资源不存在。

日志相关接口

获取Rum日志聚合信息

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取项目下的日志聚合信息

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：DescribeRumGroupLog。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
OrderBy	是	String	排序方式 desc asc（必填） 示例值：asc
StartTime	是	String	开始时间（必填） 示例值：1665993060000
Limit	是	Integer	单次查询返回的原始日志条数，最大值为100（必填） 示例值：10
Page	是	Integer	页数，第几页 示例值：1
Query	是	String	查询语句，参考控制台请求参数，语句长度最大为4096（必填） 示例值：id:100
EndTime	是	String	结束时间（必填） 示例值：1665996660000
ID	是	Integer	项目ID（必填） 示例值：100
GroupField	是	String	聚合字段 示例值：date

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"code":0,"data":[{"1737254300150":1},

参数名称	类型	描述
		{"1737254310941":1}], "msg":"","request_id":"c946f00ee5369b735c1db85c7e2fe6ff"}"
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 2312

输入示例

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

<公共请求参数>

```
{
  "OrderBy": "asc",
  "ID": 1,
  "Limit": 1,
  "GroupField": "date",
  "StartTime": "1",
  "Query": "1",
  "EndTime": "1",
  "Page": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "81a43a52-d00f-462a-82da-89841117f4ca",
    "Result": "{ \"code\":0, \"data\": [{ \"1737254300150\":1 }, { \"1737254310941\":1 } ], \"msg\": \"\", \"request_id\": \"81a43a52-d00f-462a-82da-89841117f4ca\" }"
  }
}
```

示例2 获取日志列表

输入示例

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

<公共请求参数>

```
{
  "OrderBy": "\"desc\"",
  "StartTime": "\"1\"",
}
```

```
"Limit": 10,
"Query": "\"*\"",
"EndTime": "\"20\"",
"ID": 10,
"Page": 1,
"GroupField": "date"
}
```

输出示例

```
{
  "Response": {
    "Result": "{\code\":0,\data\": [{\1737254300150\":1}, {\1737254310941\":1}], \msg\": \"\", \request_id\": \"65a8fec7-2b39-4b11-893f-3715279d235f\"}",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败

错误码	描述
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

导出Rum日志列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取项目下的日志列表（实例创建的项目下的日志列表）

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：DescribeRumLogExport。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Name	是	String	导出标识name 示例值：标识
StartTime	是	String	开始时间（必填） 示例值：1665993060000
Query	是	String	查询语句，参考控制台请求参数，语句长度最大为4096（必填） 示例值：id:100
EndTime	是	String	结束时间（必填） 示例值：1665996660000
ID	是	Integer	项目ID（必填） 示例值：100
Fields.N	否	Array of String	field条件 示例值：["msg","url"]

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"RequestId": "da2124e7-3dec-47d8-a3e2-0a84b9128571", "Result": "{\"code\":0,\"data\":{\"id\":234875,\"name\":\"132445_log\",\"message\":\"\"},\"msg\":\"\"}"}
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 1212

输入示例

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

{
  "Query": "id:1",
  "EndTime": "1",
  "Name": "",
  "StartTime": "1665994920000",
  "ID": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "763188d1-9f99-45e8-8bbf-bcdffa8f58f0",
    "Result": "{\"code\":0,\"data\":{\"id\":1,\"name\":\"132445_log\",\"message\":\"\"},\"msg\":\"\"}"
  }
}
```

示例2 获取日志列表

输入示例

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

{
  "Name": "logDemo",
  "Fields": [
    "date"
  ],
  "StartTime": "1714103013",
  "Query": "id:15",
  "EndTime": "1714103013",
  "ID": 15
}
```

输出示例

```
{
  "Response": {
    "Result": "{ \"code\":0, \"data\":{ \"id\":15, \"name\": \"132495_log\", \"message\": \"\" }, \"msg\": \"\" }",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalServerError	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。
OperationDenied	操作被拒绝。

错误码	描述
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取Rum日志导出列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取项目下的日志导出列表

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：DescribeRumLogExports。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
PageSize	是	Integer	页面大小 示例值：10
PageNum	是	Integer	页数，第几页 示例值：1
ID	是	Integer	项目ID（必填） 示例值：100

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"code":0,"data":{"page":1,"pageSize":999,"totalCount":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 xc

输入示例

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

<公共请求参数>

```
{
  "PageNum": 1,
  "ID": 234763,
  "PageSize": 999
}
```

输出示例

```
{
  "Response": {
    "RequestId": "01a45bad-d053-4af8-9f60-0c4293574f5d",
    "Result": "{\code\":0,\data\":{\page\":1,\pageSize\":999,\totalCount\":1,\totalPage\":1,\data\":[{\id\":234763,\name\":\124858_log\", \status\":\\", \message\":\任务创建成功\", \progress\":0,\url\":\\", \updateUser\":\TICLI\", \updateTime\":\2025-01-19 10:13:04\", \type\":\json\", \createUser\":\TICLI\", \createTime\":\2025-01-19 10:13:04\"]}], \msg\":\\"}"}"
  }
}
```

示例2 获取日志列表

输入示例

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

```
{
  "PageSize": 999,
  "ID": 10,
  "PageNum": 1
}
```

输出示例

```
{
  "Response": {
    "Result": "{\code\":0,\data\":{\page\":1,\pageSize\":999,\totalCount\":1,\totalPage\":1,\data\":[{\id\":10,\name\":\10_log\", \status\":\\", \message\":\任务创建成功\", \progress\":0,\url\":\\", \updateUser\":\TICLI\", \updateTime\":\2025-01-19 10:14:04\", \type\":\json\", \createUser\":\TICLI\", \createTime\":\2025-01-19 10:14:04\"]}], \msg\":\\"}",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。

错误码	描述
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取Rum日志列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取项目下的日志列表（实例创建的项目下的日志列表）

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：DescribeRumLogList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
OrderBy	是	String	排序方式 desc asc（必填） 示例值：asc
StartTime	是	String	开始时间（必填）格式为时间戳 毫秒 示例值：1665993060000
Limit	是	Integer	单次查询返回的原始日志条数，最大值为100（必填） 示例值：10
Page	是	Integer	页数，第几页 示例值：1
Query	是	String	查询语句，参考控制台请求参数，语句长度最大为4096（必填） 示例值：id:100
EndTime	是	String	结束时间（必填）格式为时间戳 毫秒 示例值：1665996660000
ID	是	Integer	项目ID（必填） 示例值：120000

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"code":0,"data":{"total":0,"limit":100,"page":1},"msg":"","request_id":"2158d1c1abbae2f1a5238786fd061c7e"}
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1 123

输入示例

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

{
  "OrderBy": "asc",
  "ID": 4,
  "Limit": 100,
  "StartTime": "1676390400000",
  "Query": "id:4",
  "EndTime": "1676476799000",
  "Page": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "a6bafa2f-3f6a-4f1a-b9a3-d1d848d34168",
    "Result": "{\\\"code\\\":0,\\\"data\\\":{\\\"total\\\":0,\\\"limit\\\":100,\\\"page\\\":1},\\\"msg\\\":\\\"\\\",\\\"request_id\\\":\\\"a6bafa2f-3f6a-4f1a-b9a3-d1d848d34168\\\"}"
  }
}
```

示例2 获取日志列表

获取日志列表

输入示例

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

{
  "OrderBy": "desc",
  "ID": 120000,
  "Limit": 100,
  "StartTime": "1665994920000",
  "Query": "id:120000",
  "EndTime": "1673417492913",
}
```

```
"Page": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "72ef03a0-01a8-443e-908a-8573fbaafcd",
    "Result": "{\"code\":0,\"data\":{\"total\":0,\"limit\":100,\"page\":1},\"msg\":\"\",\"request_id\":\"72ef03a0-01a8-443e-908a-8573fbaafcd\"}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
InternalError	内部错误。
InvalidParameter	参数错误。

错误码	描述
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取Rum分钟级日志列表

最近更新时间：2025-04-25 01:51:33

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取项目下的日志列表，分钟级

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：DescribeRumStatsLogList。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	String	开始时间（必填） 示例值：1665993060000
Limit	是	Integer	单次查询返回的原始日志条数，最大值为100（必填） 示例值：100
Query	是	String	查询语句，参考控制台请求参数，语句长度最大为4096（必填） 示例值：id:100
EndTime	是	String	结束时间（必填） 示例值：1665996660000
ID	是	Integer	项目ID（必填） 示例值：100

3. 输出参数

参数名称	类型	描述
Result	String	返回字符串 示例值：{"code":0,"data":[],"msg":""}
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 1212

输入示例

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

```
{
  "Query": "1",
  "EndTime": "1",
  "Limit": 1,
  "ID": 1,
  "StartTime": "1"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "7987e9a1-cece-4468-b104-cd239f2f6cd6",
    "Result": "{\"code\":0,\"data\":[],\"msg\":\"\"}"
  }
}
```

示例2 获取日志列表

输入示例

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

```
{
  "StartTime": "\"1\"",
  "Limit": 10,
  "Query": "\"*\"",
  "EndTime": "\"20\"",
  "ID": 10
}
```

输出示例

```
{
  "Response": {
    "Result": "Traceback (most recent call last):\n File \"/opt/app-root/lib64/python3.8/site-packages/tornado/web.py", line 1763, in _execute\n result = self.prepare()\n File \"/opt/app-root/lib64/python3.8/site-packages/tornado/web.py", line 2538, in prepare\n raise HTTPError(self._status_code)\ntornado.web.HTTPError: HTTP 404: Not Found\n",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

```
}  
  
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

业务系统和项目信息相关接口

创建 RUM 应用

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

创建 RUM 应用（归属于某个团队）

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：CreateProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Name	是	String	应用名称(不为空且最长为 200) 示例值：测试项目名
InstanceId	是	String	业务系统 ID 示例值：taw-123
Rate	是	String	项目抽样率(大于等于 0) 示例值：10
EnableURLGroup	是	Integer	是否开启聚类 示例值：1
Type	是	String	项目类型("web", "mp", "android", "ios", "node", "hippy", "weex", "viola", "rn") 示例值：web
Repo	否	String	项目对应仓库地址(可选，最长为 256) 示例值：http://github.com/xxx
URL	否	String	项目对应网页地址(可选，最长为 256) 示例值：http://www.qq.com/
Desc	否	String	应用描述(可选，最长为 1000) 示例值：这是项目描述

3. 输出参数

参数名称	类型	描述
ID	Integer	项目 id 示例值：1

参数名称	类型	描述
Key	String	项目唯一key 示例值：RlOmCVbPrKd4
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 RUM 应用

创建 RUM 应用

输入示例

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

{
  "Name": "'测试项目'",
  "InstanceId": "\"taw-123\"",
  "Repo": "'http://github.com/xxx'",
  "URL": "'http://www.qq.com'",
  "Rate": "\"10\"",
  "EnableURLGroup": 1,
  "Type": "web",
  "Desc": "'项目描述'"
}
```

输出示例

```
{
  "Response": {
    "ID": 1,
    "Key": "RlOmCVbPrKd4",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
MissingParameter	缺少参数错误。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
UnauthorizedOperation	未授权操作。

添加星标应用

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

个人用户添加星标项目

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：CreateStarProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	实例ID：taw-123 示例值：taw-123
ID	是	Integer	项目ID 示例值：1

3. 输出参数

参数名称	类型	描述
Msg	String	接口返回信息 示例值：success
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 添加星标应用

添加星标应用

输入示例

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

```
{
  "InstanceId": "taw-123",
  "ID": 1
}
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常

错误码	描述
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除 RUM 业务系统

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

删除实例，谨慎操作，不可恢复

默认接口请求频率限制：20次/秒。

推荐使用 API Explorer

</> 点击调试

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

2. 输入参数

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

参数名称	必选	类型	描述
Action	是	String	公共参数 ，本接口取值：DeleteInstance。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	需要删除的实例id 示例值：rum-abcd1234

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 RUM 业务系统

删除 RUM 业务系统

输入示例

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

{
  "InstanceId": "rum-abcd1234"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "2c56dbc9-2b99-437a-9a5e-51ed6dcba7e5"
  }
}
```

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。
FailedOperation	操作失败。
InternalError	内部错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。

错误码	描述
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

删除 RUM 应用

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

删除给定的 rum 的项目

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：DeleteProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ID	是	Integer	需要删除的项目 ID 示例值：6

3. 输出参数

参数名称	类型	描述
Msg	String	操作信息 示例值：success
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 RUM 应用

删除 RUM 应用

输入示例

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

{
```

```
"ID": 6
}
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
ResourceNotFound	资源不存在。

删除星标应用

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

删除用户名下的星标项目

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：DeleteStarProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	实例ID：-1792 示例值：-1793
ID	是	Integer	项目ID 示例值：1

3. 输出参数

参数名称	类型	描述
Msg	String	返回消息,请求成功才会返回，出现异常默认为null 示例值：success
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 删除星标应用

删除星标应用

输入示例

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

```
{
  "InstanceId": "****-1793",
  "ID": 1
}
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常

错误码	描述
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取上报量

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取项目上报量

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：DescribeDataReportCount。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
EndTime	是	Integer	结束时间 示例值：1625454840
ID	否	Integer	项目ID 示例值：1
ReportType	否	String	上报类型（custom, event, log, miniProgramData, performance, pv, speed, webvitals） 示例值：custom
InstanceId	否	String	实例ID 示例值：rum-vzysRe1w

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"4d50a7e7-cb3a-47af-a4ff-f545bfd54880","results":[{"statement_id":1,"series":[{"name":"log_url_statistics","columns":["time","allCount"],"values":[],"total":1,"offset":""}]}
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 获取上报量

输入示例

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

{
  "EndTime": 1625454840,
  "ReportType": "custom",
  "ID": 1,
  "StartTime": 1625444040
}
```

输出示例

```
{
  "Response": {
    "Result": "{\n  \"request_id\": \"4d50a7e7-cb3a-47af-a4ff-f545bfd54880\",\n  \"results\": [\n    {\n      \"statement_id\": 1,\n      \"series\": [\n        {\n          \"name\": \"log_url_statistics\",\n          \"columns\": [\n            \"time\",\n            \"allCount\"\n          ],\n          \"values\": [\n            {\n              \"total\": 1,\n              \"offset\": \"\"\n            }\n          ]\n        }\n      ]\n    }\n  ]\n}",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取应用上报抽样信息

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取应用上报抽样信息

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：DescribeProjectLimits。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	项目ID 示例值：2

3. 输出参数

参数名称	类型	描述
ProjectLimitSet	Array of ProjectLimit	上报率数组列表 示例值：[{"ID": 17,"ProjectID": 8,"ProjectInterface": "web","ReportRate": 10,"ReportType": 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 获取应用上报抽样信息

获取应用上报抽样信息

输入示例

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

{
```

```
"ProjectID": 8
}
```

输出示例

```
{
  "Response": {
    "ProjectLimitSet": [
      {
        "ID": 17,
        "ProjectID": 8,
        "ProjectInterface": "web",
        "ReportRate": 10,
        "ReportType": 1
      }
    ],
    "RequestId": "6f867613-843d-410a-8000-c74477ca48a8"
  }
}
```

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。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。

错误码	描述
FailedOperation	操作失败。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取 RUM 应用列表

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取项目列表（实例创建的团队下的项目列表）

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：DescribeProjects。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Limit	是	Integer	分页每页数目，整型 示例值：20
Offset	是	Integer	分页页码，整型 示例值：1
Filters.N	否	Array of Filter	过滤参数；demo模式传{"Name": "IsDemo", "Values":["1"]}

3. 输出参数

参数名称	类型	描述
TotalCount	Integer	列表总数 示例值：0
ProjectSet	Array of RumProject	项目列表
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取 RUM 应用列表

获取 RUM 应用列表

输入示例

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

```
{
  "Limit": 20,
  "Offset": 1
}
```

输出示例

```
{
  "Response": {
    "TotalCount": 0,
    "ProjectSet": [],
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

示例2 根据过滤条件获取 RUM 应用列表

根据过滤条件获取 RUM 应用列表

输入示例

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

```
{
  "Limit": 20,
  "Filters": [
    {
      "Values": [
        "测试项目名2",
        "测试项目名"
      ],
      "Name": "'Name'"
    }
  ],
  "Offset": 1
}
```

输出示例

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

```
"totalCount": 0,
"ProjectSet": [],
"RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
}
```

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。
InternalServerError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。

获取首页分数列表

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取首页分数列表

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：DescribeScores。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
EndTime	是	String	结束时间 示例值：2022051914
StartTime	是	String	开始时间 示例值：2020011920
ID	否	Integer	项目ID 示例值：1

3. 输出参数

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

4. 示例

示例1 获取首页Score

获取首页Score

输入示例

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


X-TC-Action: DescribeScores

<公共请求参数>

```
{
  "EndTime": "2022051914",
  "ID": 1,
  "StartTime": "2020011920"
}
```

输出示例

```
{
  "Response": {
    "ScoreSet": [
      {
        "ApiAvaliableScore": "30",
        "ApiDuration": "0",
        "ApiFail": "0",
        "ApiNum": "0",
        "ApiPerformanceScore": "5",
        "CreateTime": "",
        "JsErrorScore": "30",
        "PageDuration": "0",
        "PageError": "0",
        "PagePerformanceScore": "10",
        "PagePv": "1",
        "PageUv": "0",
        "ProjectID": 1,
        "RecordNum": 1,
        "Score": "100",
        "StaticAvaliableScore": "20",
        "StaticDuration": "210",
        "StaticFail": "0",
        "StaticNum": "9",
        "StaticPerformanceScore": "5"
      }
    ],
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

查询 RUM 业务系统信息

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

查询实例信息

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：DescribeTawInstances。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ChargeStatuses.N	否	Array of Integer	计费状态 示例值：1
ChargeTypes.N	否	Array of Integer	计费类型 示例值：1
Limit	否	Integer	分页Limit 示例值：10
Offset	否	Integer	分页Offset 示例值：1
Arealds.N	否	Array of Integer	片区Id 示例值：1
Filters.N	否	Array of Filter	过滤参数；demo模式传{"Name": "IsDemo", "Values":["1"]} 示例值：[{"Name": "Region", "Values":["ap-guangzhou"]}]

3. 输出参数

参数名称	类型	描述
InstanceSet	Array of RumInstanceInfo	实例列表
TotalCount	Integer	实例总数 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 RUM 业务系统信息

查询 RUM 业务系统信息

输入示例

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

{
  "Offset": 0,
  "Limit": 1,
  "Filters": [
    {
      "Name": "Region",
      "Values": [
        "ap-guangzhou"
      ]
    },
    {
      "Name": "InstanceType",
      "Values": [
        "1"
      ]
    }
  ]
}
```

输出示例

```
{
  "Response": {
    "InstanceSet": [
      {
        "AreaId": 1,
        "ChargeStatus": 1,
        "ChargeType": 1,
        "ClusterId": 0,
        "CreatedAt": "2024-12-17 11:12:55",
        "DataRetentionDays": 30,
        "InstanceDesc": "test333",
        "InstanceId": "****GmpbKG4",
        "InstanceName": "test-333",
        "InstanceStatus": 2,
        "InstanceType": 1,
        "Tags": [
          {
            "Key": "二级业务",

```

```
"Value": "未分配业务_200003722"
}
],
"UpdatedAt": "2024-12-18 18:22:01"
}
],
"RequestId": "8c3247ac-bafa-4e62-a73b-0b8c88e05202",
"TotalCount": 1
}
}
```

5. 开发者资源

腾讯云 API 平台

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

API Inspector

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

SDK

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

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

命令行工具

- [Tencent Cloud CLI 3.0](#)

6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

错误码	描述
AuthFailure	CAM签名/鉴权错误。
FailedOperation	操作失败。
FailedOperation.DataBaseException	数据操作异常
InternalError	内部错误。
InvalidParameter	参数错误。

修改 RUM 业务系统

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

修改 RUM 业务系统

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：ModifyInstance。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	要修改的实例id 示例值：taw-12345678
InstanceName	否	String	新的实例名称(长度最大不超过255) 示例值：实例名称
InstanceDesc	否	String	新的实例描述(长度最大不超过1024) 示例值：实例描述

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 修改实例信息

输入示例

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

```
{
  "InstanceId": "taw-12345678",
  "InstanceName": "新实例名称",
  "InstanceDesc": "新实例描述"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "411c8d9d-9cea-4d1d-906d-6f1e076ded28"
  }
}
```

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.DataBaseException	数据操作异常
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

修改 RUM 应用信息

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

修改 RUM 应用信息

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：ModifyProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ID	是	Integer	项目 id 示例值：6
Name	否	String	应用名称(可选，不为空且最长为 200 字符) 示例值：测试项目名
URL	否	String	项目网页地址(可选，最长为 256) 示例值：http://www.qq.com
Repo	否	String	项目仓库地址(可选，最长为 256) 示例值：http://github.com/xxx/yyy
InstanceId	否	String	项目需要转移到的实例 id(可选) 示例值：taw-123
Rate	否	String	项目采样率(可选) 示例值：10
EnableURLGroup	否	Integer	是否开启聚类(可选) 示例值：0
Type	否	String	项目类型(可接受值为 "web", "mp", "android", "ios", "node", "hippy", "weex", "viola", "rn") 示例值：web
Desc	否	String	应用描述(可选，最长为 1000 字符) 示例值：这是项目描述
EnableKafka	否	Integer	启动kafka配置 示例值：1
KafkaHost	否	String	kafka_host 示例值：9.134.190.4:9092

参数名称	必选	类型	描述
KafkaTopic	否	String	topic 示例值: topic
KafkaVersion	否	String	kafka_version 示例值: 1.0.0
SaslUserName	否	String	kafka_username 示例值: hawchen
SaslPassword	否	String	kafka_pwd 示例值: password
SaslMechanism	否	String	SaslMechanism 示例值: plan
SinkId	否	Integer	sink_id, 日知汇算子id 示例值: 1025

3. 输出参数

参数名称	类型	描述
Msg	String	操作信息 示例值: success
ID	Integer	项目id 示例值: 6
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 RUM 应用信息

修改 RUM 应用信息

输入示例

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

{
  "ID": 6,
  "Name": "测试项目"
}
```

输出示例

```
{
  "Response": {
    "ID": 6,
    "Msg": "success",
  }
}
```

```
"RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
}
```

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 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。
ResourceNotFound	资源不存在。

新增修改限流

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

新增修改限流

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：ModifyProjectLimit。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectID	是	Integer	项目ID 示例值：1
ProjectInterface	否	String	取值为[log speed performance webvitals pv event custom miniProgramData]其中之一 示例值：log
ReportRate	否	Integer	上报比例 10代表10% 示例值：10
ReportType	否	Integer	上报类型 1：比例 2：上报量 示例值：1
ID	否	Integer	主键ID 示例值：1

3. 输出参数

参数名称	类型	描述
Msg	String	返回信息 示例值：success
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 新增抽样

输入示例

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

```
{
  "ProjectInterface": "log",
  "ProjectID": 8,
  "ReportRate": 10,
  "ReportType": 1
}
```

输出示例

```
{
  "Response": {
    "Msg": "success",
    "RequestId": "a62c0f90-5ed0-4f3e-b285-8e4885a5ca45"
  }
}
```

示例2 新增修改限流

新增修改限流

输入示例

```
https://rum.tencentcloudapi.com/?Action=ModifyProjectLimit
&ProjectID=1
&ProjectInterface=log
&ReportRate=10
&ReportType=1
&ID=1
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Msg": "invalid ModifyProjectLimitRequest.ProjectInterface: value must be in list [log speed performance webvitals pv event custom miniProgramData]",
    "RequestId": "57e7f631-73d0-4926-b659-d4ef64818312"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。

错误码	描述
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

恢复 RUM 业务系统

最近更新时间：2025-04-25 01:51:36

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

恢复 RUM 业务系统，恢复后，用户可以正常使用和上报数据

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：ResumeInstance。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	需要恢复的实例id 示例值：rum-abcd1234
IsModifyAll	否	Boolean	修改是否包括白名单 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 恢复实例

输入示例

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

{
```



```
"InstanceId": "rum-abcd1234"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "2c56dbc9-2b99-437a-9a5e-51ed6dcba7e5"
  }
}
```

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。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
OperationDenied	操作被拒绝。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

恢复 RUM 应用

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

恢复应用使用与上报数据

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：ResumeProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectId	是	Integer	项目 id 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 恢复 RUM 应用

恢复 RUM 应用

输入示例

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

{
  "ProjectId": 1
}
```

输出示例

```
{
  "Response": {
    "RequestId": "2c56dbc9-2b99-437a-9a5e-51ed6dcba7e5"
  }
}
```

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	资源不存在。

停止RUM业务系统

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

停止实例

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：StopInstance。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
InstanceId	是	String	需要停止的实例id 示例值：rum-abcd1234
IsModifyAll	否	Boolean	修改是否包括白名单 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 停止实例

停止后，业务系统内所有应用将无法上报数据

输入示例

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

{
```

```
"InstanceId": "rum-abcd1234"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "2c56dbc9-2b99-437a-9a5e-51ed6dcba7e5"
  }
}
```

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。
FailedOperation	操作失败。
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。
ResourceInUse	资源被占用。

错误码	描述
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

停止应用上报

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

停止项目使用与上报数据

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：StopProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ProjectId	是	Integer	项目 id 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 停止应用上报

停止应用上报

输入示例

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

{
  "ProjectId": 1
}
```


输出示例

```
{
  "Response": {
    "RequestId": "2c56dbc9-2b99-437a-9a5e-51ed6dcba7e5"
  }
}
```

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。
ResourceNotFound	资源不存在。

创建 RUM 业务系统

最近更新时间：2025-04-25 01:51:37

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

创建 RUM 业务系统

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：CreateTawInstance。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
Areald	是	Integer	片区Id，(至少大于0) 示例值：1
ChargeType	是	Integer	计费类型，(1=后付费) 示例值：1
DataRetentionDays	是	Integer	数据保存时间，(至少大于0) 示例值：30
InstanceName	是	String	实例名称，(最大长度不超过255字节) 示例值：实例名称
Tags.N	否	Array of Tag	标签列表
InstanceDesc	否	String	实例描述，(最大长度不超过1024字节) 示例值：实例描述
CountNum	否	String	每天数据上报量，（不作量级限制） 示例值：100
PeriodRetain	否	String	数据存储时长计费 示例值：30
BuyingChannel	否	String	实例购买渠道("cdn" 等) 示例值：cdn
ResourcePackageType	否	Integer	预付费资源包类型(仅预付费需要) 示例值：1
ResourcePackageNum	否	Integer	预付费资源包数量(仅预付费需要) 示例值：1
InstanceType	否	Integer	实例类型 1:原web相关类型 2:app端类型 示例值：1

3. 输出参数

参数名称	类型	描述
InstanceId	String	实例Id 示例值：****49878
DealName	String	预付费订单，预付费不为null，后付费为null 示例值：****94751081
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

创建实例

输入示例

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

{
  "AreaId": 1,
  "InstanceName": "****VPH0Jn",
  "ChargeType": 1,
  "DataRetentionDays": 30,
  "InstanceDesc": "业务系统"
}
```

输出示例

```
{
  "Response": {
    "InstanceId": "****VPH0Jn",
    "DealName": "****3459432",
    "RequestId": "57f4d8a2-13b6-4ac0-9f0b-0acc7bbc1d61"
  }
}
```

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 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。
FailedOperation	操作失败。
FailedOperation.ChargeNoBalance	计费相关 – 账户余额不足。
FailedOperation.ChargeNoPayRight	计费相关 – 账户没有付费权限。
FailedOperation.ChargeParamInvalid	计费相关 – 计费参数无效。
FailedOperation.InfrastructureError	第三方依赖异常。
InternalServerError	内部错误。
InvalidParameter	参数错误。
ResourceNotFound	资源不存在。

页面性能相关接口

获取PerformancePage信息

最近更新时间：2025-05-27 01:54:41

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取PerformancePage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：DescribeDataPerformancePage。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
ID	是	Integer	项目ID 示例值：1
StartTime	是	Integer	开始时间 示例值：1625444040
EndTime	是	Integer	结束时间 示例值：1625454840
Type	是	String	pagepv：pv视图，allcount：性能视图，falls：页面加载瀑布图，samp：首屏时间，day：14天数据，nettype：网络/平台视图，performance：页面性能TOP视图，version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/：ISP视图/地区视图/浏览器视图等 示例值：pagepv
Level	否	String	日志等级 示例值：1
Isp	否	String	运营商 示例值：中国移动
Area	否	String	地区 示例值：中国
NetType	否	String	网络类型 示例值：2
Platform	否	String	平台 示例值：2

参数名称	必选	类型	描述
Device	否	String	机型 示例值：三星
VersionNum	否	String	版本 示例值：1.2.1
ExtFirst	否	String	自定义1 示例值：offline
ExtSecond	否	String	自定义2 示例值：optimized
ExtThird	否	String	自定义3 示例值：normal
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Browser	否	String	浏览器 示例值：ie
Os	否	String	操作系统 示例值：ios
Engine	否	String	浏览器引擎 示例值：ie7
Brand	否	String	品牌 示例值：苹果
From	否	String	来源页面 示例值：index.html
CostType	否	String	耗时计算方式 示例值：50
Env	否	String	环境变量 示例值：prod
NetStatus	否	String	网络状态 示例值：1
WebVitals	否	Boolean	是否返回webvitals数据 示例值：true

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"ddec0dc8-3268-4571-9ac0-566ae7537495","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0},{"statement_id":2,"total":0}, {"statement_id":3,"total":0}]}
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 22

输入示例

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

{
  "EndTime": 1658390700,
  "Type": "allcount",
  "ID": 1,
  "StartTime": 1658304300
}
```

输出示例

```
{
  "Response": {
    "RequestId": "ddec0dc8-3268-4571-9ac0-566ae7537495",
    "Result": "{\n\"request_id\":\n\"ddec0dc8-3268-4571-9ac0-566ae7537495\",\n\"results\": [{\n\"statement_id\":0,\n\"total\":0}, {\n\"statement_id\":1,\n\"total\":0}, {\n\"statement_id\":2,\n\"total\":0}, {\n\"statement_id\":3,\n\"total\":0}]]}"
  }
}
```

示例2 获取PerformancePage信息

输入示例

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

{
  "ExtSecond": "自定义2",
  "Engine": "ie2",
  "IsAbroad": "1",
  "Area": "中国",
  "NetType": "2",
  "Level": "1",
  "Os": "apple",
  "Brand": "apple",
  "Isp": "中国移动",
  "VersionNum": "版本",
  "Platform": "2",
  "ExtThird": "自定义3",
  "ExtFirst": "自定义1",
  "StartTime": 1625444040,
  "Device": "三星",
}
```

```
"From": "index.html",
"EndTime": 1625454840,
"Type": "pagepv",
"ID": 1,
"Browser": "ie"
}
```

输出示例

```
{
  "Response": {
    "Result": "{\\\"request_id\\\":\\\"ddec0dc8-3268-4571-9ac0-566ae7537495\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0},{\\\"statement_id\\\":2,\\\"total\\\":0},{\\\"statement_id\\\":3,\\\"total\\\":0}]}\",
    \"RequestId\": \"65a8fec7-2b39-4b11-893f-3715279d235f\"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败

错误码	描述
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataSetUrlStatistics信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取DescribeDataSetUrlStatistics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：DescribeDataSetUrlStatistics。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	allcount：性能视图，data：小程序，component：小程序相关，day：14天数据，nettype：网络/平台视图，performance：页面性能TOP视图，version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/：ISP视图/地区视图/浏览器视图等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果

参数名称	必选	类型	描述
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算 示例值：avg
Env	否	String	环境 示例值：prod
PackageType	否	String	获取package 示例值：loadPackage

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0}, {"statement_id":1,"total":0}, {"statement_id":2,"total":0}]}"
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 222

输入示例

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

<公共请求参数>

```
{
  "EndTime": "1625454840",
  "Type": "condition",
  "ID": "1",
  "StartTime": "1625444040"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "888d57f1-f544-4291-9a88-8638c050f208",
    "Result": "{\\\"request_id\\\":\\\"888d57f1-f544-4291-9a88-8638c050f208\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0},{\\\"statement_id\\\":2,\\\"total\\\":0}]}"
  }
}
```

示例2 获取DescribeDataSetUrlStatistics信息

输入示例

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

<公共请求参数>

```
{
  "ExtSecond": "自定义2",
  "Engine": "ie2",
  "IsAbroad": "1",
  "Area": "中国",
  "NetType": "2",
  "CostType": "avg",
  "Level": "1",
  "Os": "apple",
  "Brand": "apple",
  "Isp": "中国移动",
  "VersionNum": "版本",
  "Platform": "2",
  "ExtThird": "自定义3",
  "ExtFirst": "自定义1",
  "StartTime": "1625444040",
  "Device": "三星",
  "From": "index.html",
  "EndTime": "1625454840",
  "Type": "pagepv",
  "ID": "1",
}
```

```
"Browser": "ie"
}
```

输出示例

```
{
  "Response": {
    "Result": "{\\\"request_id\\\":\\\"65a8fec7-2b39-4b11-893f-3715279d235f\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0},{\\\"statement_id\\\":2,\\\"total\\\":0}]}\",
    \"RequestId\": \"65a8fec7-2b39-4b11-893f-3715279d235f\"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalError	内部错误。

错误码	描述
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataWebVitalsPage信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataWebVitalsPage信息，用户核心活动信息

页面加载性能之Web Vitals。性能关键点

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：DescribeDataWebVitalsPage。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：3.2.1
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Type	否	String	类型暂无 示例值：pagepv
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国

参数名称	必选	类型	描述
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算 示例值：avg
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"request_id":"2fe018e0-9e76-46cb-9ddb-cfbd6835ff19","results":[{"statement_id":0,"series":[{"name":"webvitals_page_statistics","columns":["time","LCP","FID","CLS","FCP"],"values":[[0,null,null,null,null]]}], "total":1,"offset":""}"
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WebVitalsPage信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataWebVitalsPage
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
&Isp=中国移动
&Area=中国
```



```
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "2fe018e0-9e76-46cb-9ddb-cfbd6835ff19",
    "Result": "{\n  \"request_id\": \"2fe018e0-9e76-46cb-9ddb-cfbd6835ff19\",
  \"results\": [{\n    \"statement_id\": 0,
    \"series\": [\n      {\n        \"name\": \"webvitals_page_statistics\",
        \"columns\": [\n          \"time\",
          \"LCP\",
          \"FID\",
          \"CLS\",
          \"FCP\"
        ],
        \"values\": [[0, null, null, null, null]]
      }
    ]
  },
  \"total\": 1,
  \"offset\": \"\"
}]"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

自定义测速相关接口

获取DescribeDataCustomUrl信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取DescribeDataCustomUrl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：DescribeDataCustomUrl。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	top：资源top视图，allcount：性能视图，day：14天数据，condition：条件列表，pagepv：性能视图，area：请求速度分布，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1

参数名称	必选	类型	描述
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型（1,2,3,4,5,100），1表示WIFI, 2表示2G, 3表示3G, 4表示4G, 5表示5G, 6表示6G, 100表示未知。 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外, 0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算方式 示例值：avg
Url	否	String	自定义测速的key的值 示例值：自定义测速
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "Result": {"request_id":"44202a67-c864-4d29-82f4-bfd483327e60","results":[{"statement_id":0,"total":0,"offset":""}, {"statement_id":1,"total":0,"offset":""}]}"
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DescribeDataCustomUrl信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataCustomUrl
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&CostType=avg
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "44202a67-c864-4d29-82f4-bfd483327e60",
    "Result": "{\\\"request_id\\\":\\\"44202a67-c864-4d29-82f4-bfd483327e60\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0,\\\"offset\\\":\\\"\\\"},{\\\"statement_id\\\":1,\\\"total\\\":0,\\\"offset\\\":\\\"\\\"}]}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

自定义事件相关接口

获取DescribeDataEventUrl信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取DescribeDataEventUrl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：DescribeDataEventUrl。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	allcount：性能视图，day：14天数据，condition：条件列表，ckuv：获取uv趋势，ckpv：获取pv趋势，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果

参数名称	必选	类型	描述
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
Name	否	String	筛选条件 示例值：name
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"97e9e8cc-e106-4034-889c-c8f8d3451057","results":[{"statement_id":0,"total":0,"offset":""}]}
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DescribeDataEventUrl信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataEventUrl
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
```



```
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "97e9e8cc-e106-4034-889c-c8f8d3451057",
    "Result": "{\\"request_id\\":\\"97e9e8cc-e106-4034-889c-c8f8d3451057\\",\\"results\\":[{\\"statement_id\\":0,\\"total\\":0,\\"offset\\":\\"\\"}]}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

API监控相关接口

获取DescribeDataFetchProject信息（已下线，请使用DescribeDataFetchUrl）

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataFetchProject信息。已下线，请使用DescribeDataFetchUrl

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：DescribeDataFetchProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间，示例值：1625454840 示例值：1625444040
Type	是	String	allcount：性能视图，day：14天数据，condition：条件列表，area：请求速度分布，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/：网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间，示例值：1625454840 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级（1表示白名单日志，2表示一般日志，4表示错误日志，8表示Promise 错误，16表示Ajax 请求异常，32表示JS 加载异常，64表示图片加载异常，128表示css 加载异常，256表示console.error，512表示音视频资源异常，1024表示retcode 异常，2048表示aegis report，4096表示PV日志，8192表示自定义

参数名称	必选	类型	描述
			事件，16384表示小程序 页面不存在，32768表示websocket错误，65536表示js bridge错误） 示例值：1
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型（1,2,3,4,5,100），1表示WIFI, 2表示2G, 3表示3G, 4表示4G, 5表示5G, 6表示6G, 100表示未知。 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算方式 示例值：50
Url	否	String	来源 示例值：url
Env	否	String	环境 示例值：prod
Status	否	String	httpcode响应码 示例值：200
Ret	否	String	retcode 示例值：1

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]},
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 111

输入示例

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

{
  "EndTime": "1625454840",
  "Type": "condition",
  "ID": "1",
  "StartTime": "1625444040"
}
```

输出示例

```
{
  "Response": {
    "RequestId": "9906759b-ea71-409e-b4e0-b877512713cf",
    "Result": "{\\\"request_id\\\":\\\"9906759b-ea71-409e-b4e0-b877512713cf\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"series\\\":\\\"[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":2,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":3,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":4,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":5,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":6,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":7,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":8,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0},{\\\"statement_id\\\":9,\\\"series\\\":[\\\"name\\\":\\\"fetch_project_statistics\\\",\\\"columns\\\":[\\\"key\\\",\\\"value\\\"]}],\\\"total\\\":0}]}\"
  }
}
```

示例2 获取DescribeDataFetchProject信息

输入示例

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

{
  "ExtSecond": "自定义2",
  "Engine": "ie2",
}
```

```
"IsAbroad": "1",
"Area": "中国",
"NetType": "2",
"CostType": "avg",
"Level": "1",
"Os": "apple",
"Brand": "apple",
"Isp": "中国移动",
"VersionNum": "版本",
"Platform": "2",
"ExtThird": "自定义3",
"ExtFirst": "自定义1",
"StartTime": "1625444040",
"Device": "三星",
"From": "index.html",
"EndTime": "1625454840",
"Type": "pagepv",
"ID": "1",
"Browser": "ie"
}
```

输出示例

```
{
  "Response": {
    "Result": "{\\\"request_id\\\":\\\"65a8fec7-2b39-4b11-893f-3715279d235f\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0}]}\",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataFetchUrl信息

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataFetchUrl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：DescribeDataFetchUrl。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	allcount：性能视图，pagepv：pv视图，day：14天数据，count40x：40X视图，count50x：50X视图，count5xand4x：40≤50视图，top：资源top视图，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/:网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果

参数名称	必选	类型	描述
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算方式 示例值：50
Url	否	String	来源 示例值：http
Env	否	String	环境 示例值：prod
Status	否	String	httpcode响应码 示例值：200
Ret	否	String	retcode 示例值：1
NetStatus	否	String	网络状态 示例值：1

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"d446c290-7034-47d3-8d72-46e7a692f5ac","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]}
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例11

输入示例

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

{
  "EndTime": 1658390700,
  "Type": "allcount",
  "ID": "1",
  "StartTime": 1658304300
}
```

输出示例

```
{
  "Response": {
    "RequestId": "d446c290-7034-47d3-8d72-46e7a692f5ac",
    "Result": "{\\\"request_id\\\":\\\"d446c290-7034-47d3-8d72-46e7a692f5ac\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0}]}\"
  }
}
```

示例2 333

输入示例

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

{
  "EndTime": 1625454840,
  "Type": "condition",
  "ID": "1",
  "StartTime": 1625444040
}
```

输出示例

```
{
  "Response": {
    "RequestId": "d8b037d7-08ae-401d-84cf-071e7a133c00",
    "Result": "{\\\"request_id\\\":\\\"d8b037d7-08ae-401d-84cf-071e7a133c00\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0},{\\\"statement_id\\\":1,\\\"total\\\":0}]}\"
  }
}
```

```
}  
}
```

示例3 获取DescribeDataFetchUrl信息

输入示例

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

```
{  
  "ExtSecond": "自定义2",  
  "Engine": "ie2",  
  "IsAbroad": "1",  
  "Area": "中国",  
  "NetType": "2",  
  "CostType": "avg",  
  "Level": "1",  
  "Os": "apple",  
  "Brand": "apple",  
  "Isp": "中国移动",  
  "VersionNum": "版本",  
  "Platform": "2",  
  "ExtThird": "自定义3",  
  "ExtFirst": "自定义1",  
  "StartTime": 1625444040,  
  "Device": "三星",  
  "From": "index.html",  
  "EndTime": 1625454840,  
  "Type": "pagepv",  
  "ID": "1",  
  "Browser": "ie"  
}
```

输出示例

```
{  
  "Response": {  
    "Result": "{\\"request_id\\":\\"65a8fec7-2b39-4b11-893f-3715279d235f\\",\\"results\\":[{\\"statement_id\\":0,\\"total\\":0},{\\"statement_id\\":1,\\"total\\":0}]}",  
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"  
  }  
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。

错误码	描述
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataFetchUrlInfo信息

最近更新时间：2025-04-25 01:51:38

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataFetchUrlInfo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：DescribeDataFetchUrlInfo。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	类型 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
lsp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国

参数名称	必选	类型	描述
VersionNum	否	String	版本 示例值: 1.2.1
Platform	否	String	平台 示例值: 2
ExtThird	否	String	自定义3 示例值: normal
ExtFirst	否	String	自定义1 示例值: offline
NetType	否	String	网络类型 示例值: 2
Device	否	String	机型 示例值: 三星
IsAbroad	否	String	显示是否海外,1表示海外, 0表示非海外; 默认值为空, 查询所有。 示例值: 1
Os	否	String	操作系统 示例值: ios
Browser	否	String	浏览器 示例值: ie
CostType	否	String	耗时计算方式 示例值: avg
Url	否	String	来源 示例值: http
Env	否	String	环境 示例值: prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"request_id":"1cf7b9da-6160-4176-a46d-a88bf7a152bb","results":[{"statement_id":0,"total":0,"offset":""}]}"
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DescribeDataFetchUrlInfo信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataFetchUrlInfo
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
```

```
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&CostType=avg
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Result": "{\\"request_id\\":\\"1cf7b9da-6160-4176-a46d-a88bf7a152bb\\",\\"results\\":[{\\"statement_id\\":0,\\"total\\":0,\\"offset\\":\\"\\"}]}",
    "RequestId": "1cf7b9da-6160-4176-a46d-a88bf7a152bb"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

异常分析相关接口

获取LogUrlStatistics信息

最近更新时间：2025-04-25 01:51:35

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取LogUrlStatistics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：DescribeDataLogUrlStatistics。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	analysis：异常分析，compare：异常列表对比，allcount：性能视图，condition：条件列表，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/：网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果

参数名称	必选	类型	描述
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
Env	否	String	环境区分 示例值：production
ErrorMsg	否	String	js异常信息 示例值：Script error.

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"4d50a7e7-cb3a-47af-a4ff-f545bfd54880","results":[{"statement_id":1,"series":[{"name":"log_url_statistics","columns":["time","allCount"],"values":[],"total":1,"offset":""}]}
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LogUrlStatistics信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataLogUrlStatistics
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
```

```
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Result": "{\\\"request_id\\\":\\\"4d50a7e7-cb3a-47af-a4ff-f545bfd54880\\\",\\\"results\\\":[{\\\"statement_id\\\":1,\\\"series\\\":{\\\"name\\\":\\\"log_url_statistics\\\",\\\"columns\\\":[{\\\"time\\\",\\\"allCount\\\"}],\\\"values\\\":[],\\\"total\\\":1,\\\"offset\\\":\\\"\\\"}, {\\\"statement_id\\\":2,\\\"total\\\":0,\\\"offset\\\":\\\"\\\"}]}],\\\"RequestId\\\": \"4d50a7e7-cb3a-47af-a4ff-f545bfd54880\"}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

页面访问相关接口

获取DescribeDataPvUrlStatistics信息

最近更新时间：2025-04-25 01:51:31

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataPvUrlStatistics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：DescribeDataPvUrlStatistics。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	allcount: 性能视图, day: 14天数据, vp: 性能, ckuv: uv, ckpv: pv, condition: 条件列表, nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果

参数名称	必选	类型	描述
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
Env	否	String	环境 示例值：prod
GroupByType	否	Integer	group by 参数值枚举1:1m 2:5m 3:30m 4:1h 5:1d 示例值：1
IsNewData	否	Integer	1: 查询智研 0: 走旧逻辑，已下线，勿使用 示例值：1

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"request_id":"4f93b7fc-1857-4f4d-b3ac-eb63719a8bc5","results":[{"statement_id":0,"total":0,"offset":""}]}"
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 11

1

输入示例

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

<公共请求参数>

```
{
  "StartTime": 1697078966,
  "Type": "ckuv",
  "EndTime": 1697093366,
  "ID": 120000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "4f93b7fc-1857-4f4d-b3ac-eb63719a8bc5",
    "Result": "{\\\"request_id\\\":\\\"4f93b7fc-1857-4f4d-b3ac-eb63719a8bc5\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0,\\\"offset\\\":\\\"\\\"}]}"
  }
}
```

示例2 测试

测试

输入示例

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

<公共请求参数>

```
{
  "StartTime": 1683270000,
  "Type": "ckuv",
  "EndTime": 1683356400,
  "ID": 120000
}
```

输出示例

```
{
  "Response": {
    "RequestId": "85acc203-d239-4e47-bc04-195cb09e9e83",
    "Result": "{\\\"request_id\\\":\\\"85acc203-d239-4e47-bc04-195cb09e9e83\\\",\\\"results\\\":[{\\\"statement_id\\\":0,\\\"total\\\":0,\\\"offset\\\":\\\"\\\"}]}"
  }
}
```



```
}  
}
```

示例3 获取DescribeDataPvUrlStatistics信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataPvUrlStatistics  
&ID=1  
&StartTime=1625444040  
&EndTime=1625454840  
&Type=pagepv  
&Level=1  
&Isp=中国移动  
&Area=中国  
&NetType=2  
&Platform=2  
&Device=三星  
&VersionNum=版本  
&ExtFirst=自定义1  
&ExtSecond=自定义2  
&ExtThird=自定义3  
&IsAbroad=1  
&Browser=ie  
&Os=apple  
&Engine=ie2  
&Brand=apple  
&From=index.html  
&<公共请求参数>
```

输出示例

```
{  
  "Response": {  
    "Result": "{\n\"request_id\": \"65a8fec7-2b39-4b11-893f-3715279d235f\", \"results\": [{\n\"statement_id\": 0, \"total\": 0, \"offset\": \"\"}]}",  
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"  
  }  
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

静态资源监控相关接口

获取DescribeDataStaticProject信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名：rum.tencentcloudapi.com。

获取DescribeDataStaticProject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：DescribeDataStaticProject。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	allcount：性能视图，day：14天数据，condition：条件列表，area：请求速度分布，nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图/ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级（1表示白名单日志，2表示一般日志，4表示错误日志，8表示Promise 错误，16表示Ajax 请求异常，32表示JS 加载异常，64表示图片加载异常，128表示css 加载异常，256表示console.error，512表示音视频资源异常，1024表示retcode 异常，2048表示aegis report，4096表示PV日志，8192表示自定义事件，16384表示小程序 页面不存在，32768表示websocket错误，65536表示js bridge错误） 示例值：1

参数名称	必选	类型	描述
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区 示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型（1,2,3,4,5,100），1表示WIFI, 2表示2G, 3表示3G, 4表示4G, 5表示5G, 6表示6G, 100表示未知。 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外, 0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算 示例值：avg
Url.N	否	Array of String	来源 示例值：http
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值: "{"series":[{"name":"static_project_statistics","columns":["time","allCount","successRate","avgDuration","connectTime","domainLookupTime"],"values":[[1737246360,null,null,null,null,null]],"total":1,"offset":""}]}"
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DescribeDataStaticProject信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataStaticProject
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
&Level=1
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Result": "{\\\"results\\\": [{\\\"statement_id\\\":0,\\\"series\\\": [{\\\"name\\\":\\\"static_project_statistics\\\",\\\"columns\\\": [\\\"time\\\",\\\"allCount\\\",\\\"successRate\\\",\\\"avgDuration\\\",\\\"connectTime\\\",\\\"domainLookupTime\\\"],\\\"values\\\": [[1737246360,null,null,null,null,null],[1737246420,null,null,null,null,null],[1737246480,null,null,null,null,null],[1737246540,null,null,null,null,null]]},\\\"total\\\":1,\\\"offset\\\":\\\"\\\"}}],\\\"RequestId\\\": \"65a8fec7-2b39-4b11-893f-3715279d235f\"}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataStaticResource信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataStaticResource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：DescribeDataStaticResource。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	top：资源top视图，count40x：40X视图， nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图//ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区

参数名称	必选	类型	描述
			示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算方式 示例值：avg
Url	否	String	来源 示例值：http
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"65a8fec7-2b39-4b11-893f-3715279d235f","results":[{"statement_id":0,"total":0},{"statement_id":1,"total":0}]}
RequestId	String	唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。

4. 示例

示例1 获取DescribeDataStaticResource信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataStaticResource
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
```



```
&Level=1
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&CostType=avg
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Result": "{\\"request_id\\":\\"65a8fec7-2b39-4b11-893f-3715279d235f\\",\\"results\\":[{\\"statement_id\\":0,\\"total\\":0},{\\"statement_id\\":1,\\"total\\":0}]}",
    "RequestId": "65a8fec7-2b39-4b11-893f-3715279d235f"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

获取DescribeDataStaticUrl信息

最近更新时间：2025-04-25 01:51:32

1. 接口描述

接口请求域名： rum.tencentcloudapi.com 。

获取DescribeDataStaticUrl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：DescribeDataStaticUrl。
Version	是	String	公共参数 ，本接口取值：2021-06-22。
Region	否	String	公共参数 ，此参数为可选参数。
StartTime	是	Integer	开始时间 示例值：1625444040
Type	是	String	pagepv：性能视图， nettype/version/platform/isp/region/device/browser/ext1/ext2/ext3/ret/status/from/url/env/: 网络平台视图/Version视图/设备视图/ISP视图/地区视图/浏览器视图//ext1视图等等 示例值：pagepv
EndTime	是	Integer	结束时间 示例值：1625454840
ID	是	Integer	项目ID 示例值：1
ExtSecond	否	String	自定义2 示例值：optimized
Engine	否	String	浏览器引擎 示例值：ie7
Isp	否	String	运营商 示例值：中国移动
From	否	String	来源页面 示例值：index.html
Level	否	String	日志等级 示例值：1
Brand	否	String	品牌 示例值：苹果
Area	否	String	地区

参数名称	必选	类型	描述
			示例值：中国
VersionNum	否	String	版本 示例值：1.2.1
Platform	否	String	平台 示例值：2
ExtThird	否	String	自定义3 示例值：normal
ExtFirst	否	String	自定义1 示例值：offline
NetType	否	String	网络类型 示例值：2
Device	否	String	机型 示例值：三星
IsAbroad	否	String	显示是否海外,1表示海外，0表示非海外；默认值为空，查询所有。 示例值：1
Os	否	String	操作系统 示例值：ios
Browser	否	String	浏览器 示例值：ie
CostType	否	String	耗时计算方式 示例值：avg
Url	否	String	来源 示例值：http
Env	否	String	环境 示例值：prod

3. 输出参数

参数名称	类型	描述
Result	String	返回值 示例值：{"request_id":"19f7743c-8a5d-4c41-ad6a-742a145b685d","results":[{"statement_id":0,"total":0,"offset":""}]}
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DescribeDataStaticUrl信息

输入示例

```
https://rum.tencentcloudapi.com/?Action=DescribeDataStaticUrl
&ID=1
&StartTime=1625444040
&EndTime=1625454840
&Type=pagepv
```

```
&Level=1
&Isp=中国移动
&Area=中国
&NetType=2
&Platform=2
&Device=三星
&VersionNum=版本
&ExtFirst=自定义1
&ExtSecond=自定义2
&ExtThird=自定义3
&IsAbroad=1
&Browser=ie
&Os=apple
&Engine=ie2
&Brand=apple
&From=index.html
&CostType=avg
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "RequestId": "19f7743c-8a5d-4c41-ad6a-742a145b685d",
    "Result": "{\\"request_id\\":\\"19f7743c-8a5d-4c41-ad6a-742a145b685d\\",\\"results\\":[{\\"statement_id\\":0,\\"total\\":0,\\"offset\\":\\"\\"}]}"
  }
}
```

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。
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。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
InternalServerError	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。
OperationDenied	操作被拒绝。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceNotFound.NoInstance	实例不存在
ResourceUnavailable	资源不可用。
ResourcesSoldOut	资源售罄。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。

数据结构

最近更新时间：2025-03-14 01:24:12

Filter

描述键值对过滤器，用于条件过滤查询。例如过滤ID、名称、状态等

- 若存在多个Filter时，Filter间的关系为逻辑与（AND）关系。
- 若同一个Filter存在多个Values，同一Filter下Values间的关系为逻辑或（OR）关系。

被如下接口引用：DescribeProjects, DescribeTawInstances。

名称	类型	必选	描述
Values	Array of String	否	一个或者多个过滤值。 示例值：ap-guangdong,ap-shenzhen
Name	String	否	过滤键的名称。 示例值：Zone

Kafka

旁路kafka配置

被如下接口引用：DescribeProjects。

名称	类型	描述
EnableKafka	Integer	1: 开启 0: 关闭 示例值：1
KafkaHost	String	host地址 示例值：9.134.190.4:9092
KafkaTopic	String	topic 示例值：hello
KafkaVersion	String	版本 示例值：1.0.0
SaslUserName	String	username 示例值：hachen
SaslPassword	String	password 示例值：12345214
SaslMechanism	String	ssl 示例值：plain
SinkId	Integer	默认算子id为0新增算子 一旦算子新增成功会返回正确的算子id值 示例值：0

ProjectLimit

项目接口限制类型

被如下接口引用：DescribeProjectLimits。

名称	类型	必选	描述
ProjectInterface	String	是	接口 示例值：web
ReportRate	Integer	是	上报率 示例值：10
ReportType	Integer	是	上报类型 1：上报率 2：上报量限制 示例值：1
ID	Integer	否	主键ID 示例值：8
ProjectID	Integer	否	项目ID 示例值：8

ReleaseFile

发布文件列表(SOURCEMAP)

被如下接口引用：CreateReleaseFile, DescribeReleaseFiles。

名称	类型	必选	描述
Version	String	否	文件版本 示例值：0.0.1
FileKey	String	否	文件唯一 key 示例值：DFFDSGGSDFFDS
FileName	String	否	文件名 示例值：a.js.map
FileHash	String	否	文件哈希值 示例值：asdfsfdsg/f/g
ID	Integer	否	文件 id 示例值：1

RumAreaInfo

Rum片区信息

被如下接口引用：DescribeTawAreas。

名称	类型	描述
Areald	Integer	片区Id 示例值：1
AreaStatus	Integer	片区状态(1=有效，2=无效) 示例值：1
AreaName	String	片区名称 示例值：中国大陆
AreaKey	String	片区Key 示例值：area-cn-dl
AreaRegionID	String	地域码表 id 示例值：1

名称	类型	描述
AreaRegionCode	String	地域码表 code 如 ap-xxx (xxx 为地域词) 示例值：ap-guangzhou
AreaAbbr	String	地域缩写 示例值：gz

RumInstanceInfo

Rum实例信息

被如下接口引用：DescribeTawInstances。

名称	类型	描述
InstanceStatus	Integer	实例状态(1=创建中, 2=运行中, 3=异常, 4=重启中, 5=停止中, 6=已停止, 7=已删除) 示例值：2
Areald	Integer	片区Id 示例值：1
Tags	Array of Tag	标签列表
InstanceId	String	实例Id 示例值：taw-12345678
ClusterId	Integer	集群Id 示例值：1
InstanceDesc	String	实例描述 示例值：taw-user
ChargeStatus	Integer	计费状态(1=使用中, 2=已过期, 3=已销毁, 4=分配中, 5=分配失败) 示例值：1
ChargeType	Integer	计费类型(1=免费版, 2=预付费, 3=后付费) 示例值：1
UpdatedAt	String	更新时间 示例值：2021-01-01 00:00:00
DataRetentionDays	Integer	数据保留时间(天) 示例值：30
InstanceName	String	实例名称 示例值：taw-user
CreatedAt	String	创建时间 示例值：2021-01-01 00:00:00
InstanceType	Integer	实例类型 1:原web相关类型 2:app端类型 示例值：1

RumProject

Rum 项目信息

被如下接口引用：DescribeProjects。

名称	类型	描述
Name	String	项目名 示例值：测试项目

名称	类型	描述
Creator	String	创建者 id 示例值：ouyang
InstanceId	String	实例 id 示例值：tam-123
Type	String	项目类型 示例值：log
CreateTime	Timestamp	创建时间 示例值：2020-04-26 00:00:00
Repo	String	项目仓库地址 示例值：http://github.com/xxx/yyy
URL	String	项目网址地址 示例值：http://cloud.tencent.com/index.html
Rate	String	项目采样频率 示例值：12
Key	String	项目唯一key（长度 12 位） 示例值：hjkysfkey12
EnableURLGroup	Integer	是否开启url聚类 示例值：1
InstanceName	String	实例名 示例值：tamIns
ID	Integer	项目 ID 示例值：1
InstanceKey	String	实例 key 示例值：tam-key
Desc	String	项目描述 示例值：这是一个项目描述
IsStar	Integer	是否星标 1:是 0:否 示例值：1
ProjectStatus	Integer	项目状态(1 创建中, 2 运行中, 3 异常, 4 重启中, 5 停止中, 6 已停止, 7 销毁中, 8 已销毁) 示例值：2
AccessPoint	String	日志接入点，用户忽略。 示例值：RUM_LOG_1
Kafka	Kafka	kafka旁路配置信息 示例值：Kafka

RumPvInfo

rum 日志对象

被如下接口引用：DescribePvList。

名称	类型	描述
ProjectId	Integer	项目ID 示例值：1

名称	类型	描述
Pv	String	pv访问量 示例值：813
CreateTime	String	时间 示例值：20201102

RumUvInfo

RumUv 访问量

被如下接口引用：DescribeUvList。

名称	类型	描述
ProjectId	Integer	项目ID 示例值：1
Uv	String	uv访问量 示例值：222
CreateTime	String	创建时间 示例值：20201102

ScoreInfo

project Score分数实体

被如下接口引用：DescribeScores。

名称	类型	描述
StaticDuration	String	duration 示例值：1
PagePv	String	pv 示例值：10
ApiFail	String	失败 示例值：1
ApiNum	String	请求 示例值：12
StaticFail	String	fail 示例值：12
ProjectID	Integer	项目id 示例值：1
PageUv	String	uv 示例值：12
ApiDuration	String	请求次数 示例值：12
Score	String	项目总分 示例值：100
PageError	String	error 示例值：10

名称	类型	描述
StaticNum	String	num 示例值：12
RecordNum	Integer	num 示例值：1
PageDuration	String	Duration 示例值：10
CreateTime	String	时间 示例值：20230111
PagePerformanceScore	String	页面性能评分 示例值：10
JsErrorScore	String	js错误评分 示例值：30
ApiPerformanceScore	String	API性能评分 示例值：5
ApiAvaliableScore	String	API可用性评分 示例值：30
StaticPerformanceScore	String	静态资源性能评分 示例值：5
StaticAvaliableScore	String	静态资源可用性评分 示例值：20

Tag

标签

被如下接口引用：CreateTawInstance, DescribeTawInstances。

名称	类型	必选	描述
Key	String	是	标签key 示例值：key
Value	String	是	标签value 示例值：value

Whitelist

白名单

被如下接口引用：DescribeWhitelists。

名称	类型	描述
Remark	String	备注 示例值：备注
InstanceId	String	实例ID 示例值：taw-123
Ttl	String	截止时间 示例值：20200527221900

名称	类型	描述
ID	String	白名单自增ID 示例值：18
WhitelistUin	String	业务唯一标识 示例值：****563044
CreateUser	String	创建者ID 示例值：T. hao
Aid	String	aid标识 示例值：****237
CreateTime	String	创建时间 示例值：20200527221900

错误码

最近更新时间：2024-12-20 01:49:40

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 <code>Authorization</code> 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。
IplnBlacklist	IP 地址在黑名单中。

错误码	说明
IpNotInWhitelist	IP 地址不在白名单中。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数。
NoSuchProduct	产品不存在
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
RequestLimitExceeded.GlobalRegionUinLimitExceeded	主账号超过频率限制。
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。
FailedOperation.ChargeNoBalance	计费相关 – 账户余额不足。
FailedOperation.ChargeNoPayRight	计费相关 – 账户没有付费权限。
FailedOperation.ChargeParamInvalid	计费相关 – 计费参数无效。
FailedOperation.ClsCallFail	CLS调用失败
FailedOperation.DataBaseException	数据操作异常
FailedOperation.InfrastructureError	第三方依赖异常。
OperationDenied	操作被拒绝。
ResourceNotFound.NoInstance	实例不存在
ResourcesSoldOut	资源售罄。

