

云日志服务

API 参考

文档版本 01
发布日期 2025-02-12



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

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

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

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

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

目录

1 使用前必读	1
2 API 概览	3
3 如何调用 API	5
3.1 构造请求	5
3.2 认证鉴权	8
3.3 返回结果	10
4 接口调用示例	12
5 应用示例	13
5.1 示例 1: 创建日志组	13
5.2 示例 2: 查询账号下所有日志组	13
5.3 示例 3: 创建日志流	14
5.4 示例 4: 查询指定日志组下的所有日志流	15
5.5 示例 5: 创建 OBS 转储	15
6 API 说明	17
6.1 主机组管理	17
6.1.1 查询主机信息	17
6.1.2 查询主机组	27
6.1.3 创建主机组	36
6.1.4 删除主机组	44
6.1.5 修改主机组	50
6.2 日志组管理	58
6.2.1 创建日志组	58
6.2.2 查询账号下所有日志组	66
6.2.3 删除日志组	69
6.2.4 修改日志组	73
6.3 日志流管理	80
6.3.1 创建日志流	80
6.3.2 查询指定日志组下的所有日志流	88
6.3.3 查询日志流信息	92
6.3.4 删除日志流	96
6.3.5 创建日志流索引	100

6.3.6 修改日志流.....	110
6.4 日志管理.....	117
6.4.1 统计 top n 的日志组或日志流流量.....	117
6.4.2 按时间段统计查询资源.....	124
6.4.3 查询日志.....	131
6.4.4 查询结构化日志（即将下线不推荐使用）.....	160
6.4.5 查询结构化日志（新版）.....	173
6.4.6 查询日志直方图.....	190
6.4.7 查询上下文日志.....	196
6.4.8 取消收藏.....	205
6.4.9 创建日志收藏.....	210
6.5 日志接入.....	217
6.5.1 新建跨账号日志接入.....	217
6.5.2 查询日志接入.....	227
6.5.3 创建日志接入.....	246
6.5.4 删除日志接入.....	283
6.5.5 修改日志接入.....	299
6.6 日志转储.....	327
6.6.1 创建日志转储（旧版）.....	327
6.6.2 创建日志转储（新版）.....	332
6.6.3 删除日志转储.....	354
6.6.4 更新日志转储.....	363
6.6.5 查询日志转储.....	385
6.6.6 注册 DMS kafka 实例.....	394
6.7 超额采集.....	401
6.7.1 关闭超额采集开关.....	401
6.7.2 打开超额采集开关.....	403
6.8 云端结构化配置.....	406
6.8.1 创建结构化配置（推荐）.....	406
6.8.2 修改结构化配置（推荐）.....	429
6.8.3 删除结构化配置.....	451
6.8.4 查询结构化配置.....	457
6.8.5 查询结构化模板简略列表.....	465
6.8.6 查询结构化模板.....	468
6.8.7 创建结构化配置（不推荐）.....	476
6.8.8 修改结构化配置（不推荐）.....	501
6.9 AOM 容器日志接入 LTS.....	524
6.9.1 创建接入规则.....	524
6.9.2 修改接入规则.....	536
6.9.3 删除接入规则.....	548
6.9.4 查询所有接入规则.....	552
6.9.5 查询单个接入规则.....	560

6.10 告警主题.....	568
6.10.1 查询 SMN 主题.....	568
6.11 消息模板管理.....	571
6.11.1 创建消息模板.....	571
6.11.2 修改消息模板.....	585
6.11.3 查询消息模板列表.....	600
6.11.4 删除消息模板.....	606
6.11.5 查询单个消息模板.....	611
6.11.6 预览消息模板邮件格式.....	616
6.12 SQL 告警规则.....	621
6.12.1 创建 SQL 告警规则.....	622
6.12.2 修改 SQL 告警规则.....	634
6.12.3 查询 SQL 告警规则.....	651
6.12.4 删除 SQL 告警规则.....	658
6.12.5 切换告警规则状态.....	660
6.13 关键词告警规则.....	668
6.13.1 创建关键词告警规则.....	668
6.13.2 修改关键词告警规则.....	680
6.13.3 查询关键词告警规则.....	697
6.13.4 删除关键词告警规则.....	705
6.14 告警列表.....	706
6.14.1 查询活动或历史告警列表.....	707
6.14.2 删除活动告警.....	716
6.15 标签管理.....	723
6.15.1 创建标签.....	723
6.16 仪表盘管理.....	729
6.16.1 创建仪表盘分组.....	729
6.16.2 创建仪表盘.....	735
6.17 日志流图表.....	741
6.17.1 查询日志流图表.....	741
6.18 快速查询.....	744
6.18.1 查询用户历史 sql.....	745
6.18.2 添加快速查询.....	751
6.18.3 获取快速查询.....	757
6.18.4 删除快速查询.....	761
6.18.5 查询日志组下所有快速查询.....	767
6.19 多账号日志汇聚.....	773
6.19.1 获取日志汇聚开关.....	773
6.19.2 修改日志汇聚开关.....	775
6.19.3 获取组织成员汇聚配置.....	777
6.19.4 获取组织成员日志组日志流.....	780
6.19.5 更新汇聚配置.....	783

7 权限和授权项	790
8 附录	793
8.1 状态码	793
8.2 错误码	793
8.3 获取账号 ID、项目 ID、日志组 ID、日志流 ID	795
8.4 OBS 转储时区表	796

1 使用前必读

欢迎使用云日志服务（Log Tank Service，简称LTS）。LTS用于收集来自主机和云服务的日志数据，通过海量日志数据的分析与处理，可以将云服务和应用程序的可用性和性能最大化，为您提供一个实时、高效、安全的日志处理能力，帮助您快速高效地进行实时决策分析、设备运维管理、用户业务趋势分析等。

您可以使用本文档提供API对云日志服务进行相关操作，如创建日志组、删除日志组、日志接入、日志转储等。支持的全部操作请参见[API概览](#)。

在调用云日志服务API之前，请确保已经充分了解云日志服务相关概念，详细信息请参见[产品介绍](#)。

云日志服务提供了REST（Representational State Transfer）风格API，支持您通过HTTPS请求调用，调用方法请参见[如何调用API](#)。

同时云日志服务还提供多种编程语言的SDK供您使用，SDK的使用方法请参见[使用SDK接入LTS](#)。

终端节点

终端节点即调用API的[请求地址](#)，不同服务在不同区域的终端节点不同，您可以从[地区和终端节点](#)中查询所有服务的终端节点。

基本概念

- 账号
用户注册时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用该用户进行日常管理工作。
- 用户
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
在[我的凭证](#)下，您可以查看账号ID和IAM用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域（Region）
从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属

Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。

详情请参见[区域和可用区](#)。

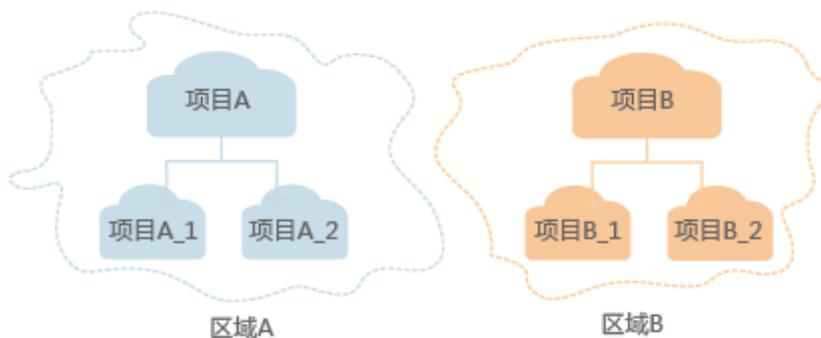
- 可用区（AZ，Availability Zone）

一个可用区是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

- 项目

区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中创建资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中资源，使得资源的权限控制更加精确。

图 1-1 项目隔离模型



同样在[我的凭证](#)下，您可以查看项目ID。

- 企业项目

企业项目是项目的升级版，针对企业不同项目间资源的分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。

关于企业项目ID的获取及企业项目特性的详细信息，请参见《[企业管理用户指南](#)》。

2 API 概览

云日志服务所提供的接口为扩展接口。通过使用云日志服务所提供的接口，您可以完整的使用云日志服务的基本功能。例如查询API版本号、创建、查询、删除日志组和日志流。

云日志服务提供的具体API如[表1](#)所示。

表 2-1 接口说明

子类型	说明
主机组管理	LTS API的主机组管理接口，用来创建、查询、删除主机组。
日志组管理	LTS API的日志组管理接口，用来创建、查询、删除日志组。
日志流管理	LTS API的日志流管理接口，用来创建、查询、删除日志流。
日志管理	LTS API的日志管理接口，用来查询日志内容。
日志接入	LTS API的日志接入接口，用来配置日志上报。
日志转储	LTS API的日志转储接口，用于将指定的一个或多个日志流的日志转储到其他服务。
超额采集	LTS API的超额采集接口，用来设置打开、关闭超额采集开关。
云端结构化配置	LTS API的结构化配置接口，用来创建、查询、修改、删除结构化配置等。
AOM容器日志接入LTS	LTS API的日志接入接口，用来创建、查询、修改、删除AOM容器日志接入LTS的接入规则等。
告警主题	LTS API的告警主题接口，用来查询SMN主题。
消息模板管理	LTS API的消息模板接口，用来创建、修改、删除、查询消息模板等。
SQL告警规则	LTS API的SQL告警接口，用来创建、查询、修改、删除SQL告警。
关键词告警规则	LTS API的关键词告警接口，用来创建、查询、修改、删除关键词告警。

子类型	说明
告警列表	LTS API的告警列表接口，用来查询、删除告警。
标签管理	LTS API的标签管理接口，用来给日志组、日志流、主机组等添加标签。
仪表盘管理	LTS API的仪表盘接口，用来创建仪表盘。
日志流图表	LTS API的日志流图表接口，用来查询日志流图表。
快速查询	LTS API的快速查询接口，用来创建、查询、删除快速查询。
多账号日志汇聚	LTS API的多账号日志汇聚接口，用来获取、修改日志汇聚开关、获取组织成员汇聚等。

3 如何调用 API

3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM的[管理员创建IAM用户](#)说明如何调用API。

您还可以通过这个视频教程了解如何构造请求调用API：<https://bbs.huaweicloud.com/videos/102987>。

请求 URI

请求URI由如下部分组成。

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

- **URI-scheme**：表示用于传输请求的协议，当前所有API均采用HTTPS协议。
- **Endpoint**：指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从[地区和终端节点](#)获取。
例如IAM服务在“华北-北京一”区域的Endpoint为“iam.cn-north-1.myhuaweicloud.com”。
- **resource-path**：资源路径，也即API访问路径。从具体API的URI模块获取，例如“管理员创建IAM用户”接口的resource-path为“/v3.0/OS-USER/users”。
- **query-string**：查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。

例如您需要创建IAM用户，由于IAM为全局服务，则使用任一区域的Endpoint（比如“华北-北京一”区域的Endpoint：“iam.cn-north-1.myhuaweicloud.com”，并在[管理员创建IAM用户](#)的URI部分找到resource-path（/v3.0/OS-USER/users），拼接起来如下所示。

```
https://iam.cn-north-1.myhuaweicloud.com/v3.0/OS-USER/users
```

图 3-1 URI 示意图



说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，同一个服务的Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

- **GET**：请求服务器返回指定资源。
- **PUT**：请求服务器更新指定资源。
- **POST**：请求服务器新增资源或执行特殊操作。
- **DELETE**：请求服务器删除指定资源，如删除对象等。
- **HEAD**：请求服务器资源头部。
- **PATCH**：请求服务器更新资源的部分内容。当资源不存在的时候，PATCH可能会去创建一个新的资源。

在“获取用户Token”的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3.0/OS-USER/users
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

详细的公共请求消息头字段请参见[表3-1](#)。

表 3-1 公共请求消息头

名称	描述	是否必选	示例
Host	请求的服务器信息，从服务API的URL中获取。值为hostname[:port]。端口缺省时使用默认的端口，https的默认端口为443。	否 使用AK/SK认证时该字段必选。	code.test.com or code.test.com:443

名称	描述	是否必选	示例
Content-Type	消息体的类型（格式）。推荐用户使用默认值 application/json，有其他取值时会在具体接口中专门说明。	是	application/json
Content-Length	请求body长度，单位为Byte。	否	3495
X-Project-Id	project id，项目编号。请参考 获取账号ID、项目ID、日志组ID、日志流ID 章节获取项目编号。	否 如果是专属云场景采用AK/SK认证方式的接口请求或者多project场景采用AK/SK认证的接口请求，则该字段必选。	e9993fc787d94b6c886cb aa340f9c0f4
X-Auth-Token	用户Token。 用户Token也就是调用 获取用户Token 接口的响应值，该接口是唯一不需要认证的接口。 请求响应成功后在响应消息头（Headers）中包含的“X-Subject-Token”的值即为Token值。	否 使用Token认证时该字段必选。	注：以下仅为Token示例片段 MIIPAgYJKoZlhvcNAQcCo ...ggg1BBIIlNPXsidG9rZ

📖 说明

API同时支持使用AK/SK认证，AK/SK认证是使用SDK对请求进行签名，签名过程会自动往请求中添加Authorization（签名认证信息）和X-Sdk-Date（请求发送的时间）请求头。

AK/SK认证的详细说明请参见[认证鉴权](#)的“AK/SK认证”。

对于[管理员创建IAM用户](#)接口，使用AK/SK方式认证时，添加消息头后的请求如下所示。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3.0/OS-USER/users
```

请求消息体（可选）

该部分可选。请求消息体通常以结构化格式（如JSON或XML）发出，与请求消息头中Content-Type对应，传递除请求消息头之外的内容。若请求消息体中的参数支持中文，则中文字符必须为UTF-8编码。

每个接口的请求消息体内容不同，也并不是每个接口都需要有请求消息体（或者说消息体为空），GET、DELETE操作类型的接口就不需要消息体，消息体具体内容需要根据具体接口而定。

对于[管理员创建IAM用户](#)接口，您可以从接口的请求部分看到所需的请求参数及参数说明。将消息体加入后的请求如下所示，加粗的斜体字段需要根据实际值填写，

- **accountid**为IAM用户所属的账号ID。
- **username**为要创建的IAM用户名。
- **email**为IAM用户的邮箱。
- *********为IAM用户的登录密码。

📖 说明

scope参数定义了Token的作用域，下面示例中获取的Token仅能访问project下的资源。您还可以设置Token的作用域为某个账号下所有资源或账号的某个project下的资源，详细定义请参见[获取用户Token](#)。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****
```

```
{
  "user": {
    "domain_id": "accountid",
    "name": "username",
    "password": "*****",
    "email": "email",
    "description": "IAM User Description"
  }
}
```

到此为止，一个API请求所需要的内容已经准备完成，您可以使用[curl](#)、[Postman](#)或直接编写代码等方式发送请求调用API。对于获取用户Token接口，返回的响应消息头中“x-subject-token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key）加密调用请求。推荐使用AK/SK认证，其安全性比Token认证要高。
- Token认证：通过Token认证调用请求。

AK/SK 认证

📖 说明

- AK/SK 签名认证方式仅支持消息体大小 12MB 以内，12MB 以上的请求请使用 Token 认证。
- AK/SK 既可以使用永久访问密钥中的 AK/SK，也可以使用临时访问密钥中的 AK/SK，但使用临时访问密钥的 AK/SK 时需要额外携带 “X-Security-Token” 字段，字段值为临时访问密钥的 security_token。

AK/SK 认证就是使用 AK/SK 对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK (Access Key ID)：访问密钥 ID。与私有访问密钥关联的唯一标识符；访问密钥 ID 和私有访问密钥一起使用，对请求进行加密签名。
- SK (Secret Access Key)：与访问密钥 ID 结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用 AK/SK 认证时，您可以基于签名算法使用 AK/SK 对请求进行签名，也可以使用专门的签名 SDK 对请求进行签名。详细的签名方法和 SDK 使用方法请参见 [API 签名指南](#)。

须知

签名 SDK 只提供签名功能，与服务提供的 SDK 不同，使用时请注意。

您也可以通过这个视频教程了解 AK/SK 认证的使用：<https://bbs.huaweicloud.com/videos/100697>。

Token 认证

📖 说明

- Token 的有效期为 24 小时，需要使用一个 Token 鉴权时，可以先缓存起来，避免频繁调用。
- 使用 Token 前请确保 Token 离过期有足够的时间，防止调用 API 的过程中 Token 过期导致调用 API 失败。

Token 在计算机系统中代表令牌（临时）的意思，拥有 Token 就代表拥有某种权限。Token 认证就是在调用 API 的时候将 Token 加到请求消息头，从而通过身份认证，获得操作 API 的权限。

Token 可通过调用 [获取用户 Token](#) 接口获取，调用本服务 API 需要 project 级别的 Token，即调用 [获取用户 Token](#) 接口时，请求 body 中 auth.scope 的取值需要选择 project，如下所示。

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    }
  }
}
```

```
},  
  "scope": {  
    "project": {  
      "name": "xxxxxxxx"  
    }  
  }  
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFJ...”，则调用接口时将“X-Auth-Token: ABCDEFJ...”加到请求消息头即可，如下所示。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3.0/OS-USER/users  
Content-Type: application/json  
X-Auth-Token: ABCDEFJ....
```

您还可以通过这个视频教程了解如何使用Token认证：<https://bbs.huaweicloud.com/videos/101333>。

3.3 返回结果

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，详情请参见[状态码](#)。

对于[管理员创建IAM用户](#)接口，如果调用后返回状态码为“201”，则表示请求成功。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于[管理员创建IAM用户](#)接口，返回如[图3-2](#)所示的消息头。

图 3-2 管理员创建 IAM 用户响应消息头

```
"X-Frame-Options": "SAMEORIGIN",  
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fceac8a",  
"Transfer-Encoding": "chunked",  
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",  
"Server": "api-gateway",  
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",  
"X-Content-Type-Options": "nosniff",  
"Connection": "keep-alive",  
"X-Download-Options": "noopen",  
"X-XSS-Protection": "1; mode=block;",  
"X-IAM-Trace-Id": "token_█_null_af2953f2bcc67a42325a69a19e6c32a2",  
"Date": "Tue, 21 May 2024 09:03:40 GMT",  
"Content-Type": "application/json; charset=utf8"
```

响应消息体（可选）

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于**管理员创建IAM用户**接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "user": {
    "id": "c131886aec...",
    "name": "IAMUser",
    "description": "IAM User Description",
    "areacode": "",
    "phone": "",
    "email": "****@***.com",
    "status": null,
    "enabled": true,
    "pwd_status": false,
    "access_mode": "default",
    "is_domain_owner": false,
    "xuser_id": "",
    "xuser_type": "",
    "password_expires_at": null,
    "create_time": "2024-05-21T09:03:41.000000",
    "domain_id": "d78cbac1.....",
    "xdomain_id": "30086000.....",
    "xdomain_type": "",
    "default_project_id": null
  }
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
  "error_msg": "Request body is invalid.",
  "error_code": "IAM.0011"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

4 接口调用示例

本节通过调用LTS的API创建日志组。

📖 说明

通过IAM获取到的Token有效期为24小时，需要使用同一个Token鉴权时，可以先将Token缓存，避免频繁调用。

涉及 API

当您使用Token认证方式完成认证鉴权时，需要获取用户Token并在调用接口时增加“X-Auth-Token”到业务接口请求消息头中。

- IAM获取token的API。
- LTS创建日志组的API。

具体步骤

1. Token认证，具体操作请参考[构造请求](#)。
2. 发送“POST /v2/{project_id}/groups”。
3. 在Request Header中增加“Content-Type”和“X-Auth-Token”。
4. 在Request Body中传入参数如下：

```
POST /v2/{project_id}/groups
{
  "log_group_name":"test001", //日志组名称(必填, String)
  "ttl_in_days":"7", // 日志过期时间(默认, int)
}
```

请求响应成功后，返回已创建日志组的信息。

```
{
  "log_group_id":"2a0089e4-3001-11e9-9e9d-286ed48xxx", //日志组id (String)
}
```

若请求失败，则会返回错误码及对应的错误信息说明，详情请参考：[错误码](#)。

5 应用示例

5.1 示例 1：创建日志组

场景描述

本章以创建一个LTS的日志组为例。

涉及其它云服务接口

无。

创建日志组

- 接口相关信息

URI格式：POST /v2/{project_id}/groups

- 请求示例

POST https://{endpoint}/v2/{project_id}/groups

{endpoint}信息请从[终端节点](#)获取。

Body:

```
{
  "log_group_name": "lts-group-01nh",
  "ttl_in_days": 7
}
```

- 响应示例

```
{
  "log_group_id": "string"
}
```

5.2 示例 2：查询账号下所有日志组

场景描述

本章以查询账号下所有日志组信息为例。

涉及其它云服务接口

无。

查询日志组

- 接口相关信息

URI格式: GET /v2/{project_id}/groups

- 请求示例

GET https://{endpoint}/v2/{project_id}/groups

{endpoint}信息请从[终端节点](#)获取。

- 响应示例

```
{"log_groups": [{"creation_time": number, "log_group_name": "string", "log_group_id": "string", "ttl_in_days": 1}]}
```

5.3 示例 3: 创建日志流

场景描述

本章以创建某个指定日志组下的日志流为例。

涉及其它云服务接口

无。

创建日志流

- 接口相关信息

URI格式: POST /v2/{project_id}/groups/{log_group_id}/streams

- 请求示例

POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams

{endpoint}信息请从[终端节点](#)获取。

Body:

```
{  
  "log_stream_name": "lts-stream-02kh"  
}
```

- 响应示例

```
{  
  "log_stream_id": "string"  
}
```

5.4 示例 4：查询指定日志组下的所有日志流

场景描述

本章以查询指定日志组下的所有日志流信息为例。

涉及其它云服务接口

无。

查询日志流

- 接口相关信息

URI格式：GET /v2/{project_id}/groups/{log_group_id}/streams

- 请求示例

GET https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams

{endpoint}信息请从[终端节点](#)获取。

- 响应示例

```
{
  "log_streams": [ {
    "creation_time": number,
    "log_stream_name": "string",
    "log_stream_id": "string",
    "filter_count": number
  } ]
}
```

5.5 示例 5：创建 OBS 转储

场景描述

本章以将指定的一个或多个日志流的日志转储到OBS服务为例。

涉及其它云服务接口

创建OBS转储时，需要查询用户的桶列表：

[获取桶列表](#)：确定将要配置转储的桶名称。

创建 OBS 转储

- 接口相关信息

URI格式：POST /v2/{project_id}/log-dump/obs

- 请求示例

POST https://{endpoint}/v2/{project_id}/log-dump/obs

{endpoint}信息请从[终端节点](#)获取。

Body:

```
{
  "log_group_id": "d9dba9f3-xxxx-48bd-xxxx-xxxxa24a8053",
  "log_stream_ids": ["45e7f609-xxxx-4cd3-835b-xxxx4a124718"],
  "obs_bucket_name": "lts-test",
  "type": "cycle",
  "storage_format": "RAW",
  "switch_on": "true",
  "prefix_name": "fileprefixname",
  "dir_prefix_name": "dirprefixname",
  "period": 5,
  "period_unit": "min"
}
```

● 响应示例

```
{
  "log_dump_obs_id" : "45fdc36b-xxxx-4567-xxxx-559xxxxdf968"
}
```

6 API 说明

本章节所介绍的所有接口URI，请在调用时务必区分大小写。

6.1 主机组管理

6.1.1 查询主机信息

功能介绍

查询主机列表

调用方法

请参见[如何调用API](#)。

URI

POST /v3/{project_id}/lts/host-list

表 6-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：1 最大长度：64

请求参数

表 6-2 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1 最大长度：10000
Content-Type	是	String	该字段填为：application/json;charset=utf8。 最小长度：30 最大长度：30

表 6-3 请求 Body 参数

参数	是否必选	参数类型	描述
host_id_list	否	Array of strings	主机ID列表。可以根据主机ID列表进行批量过滤 最小长度：36 最大长度：36
filter	是	GetHostListFilter object	查询主机信息过滤参数

表 6-4 GetHostListFilter

参数	是否必选	参数类型	描述
host_name_list	否	Array of strings	主机名称列表。可以根据主机名称列表，进行批量过滤 最小长度：1 最大长度：128
host_ip_list	否	Array of strings	主机IP列表。可以根据主机IP列表，进行批量过滤。 最小长度：1 最大长度：16

参数	是否必选	参数类型	描述
host_status	否	String	主机状态。可以根据主机状态进行过滤。 uninstall:未安装 running:运行 offline:离线 error:异常 plugin error:插件错误 installing:安装中 install-fail:安装失败 upgrading:升级中 upgrading-transient:升级中 upgrade failed:升级失败 upgrade-fail:升级失败 uninstalling:卸载中 uninstalling-transient:卸载中 authentication error:鉴权失败 枚举值: <ul style="list-style-type: none">• uninstall• running• offline• error• plugin error• installing• install-fail• upgrading• upgrading-transient• upgrade failed• upgrade-fail• uninstalling• uninstalling-transient• authentication error
host_version	否	String	主机版本。可以根据主机版本进行过滤。 最小长度: 1 最大长度: 16

响应参数

状态码：200

表 6-5 响应 Body 参数

参数	参数类型	描述
result	Array of GetHostListInfo objects	主机列表
total	Long	主机信息总数量 最小值：0 最大值：10000

表 6-6 GetHostListInfo

参数	参数类型	描述
host_id	String	主机ID 最小长度：36 最大长度：36
host_ip	String	主机IP 最小长度：1 最大长度：16
host_name	String	主机名称 最小长度：1 最大长度：128

参数	参数类型	描述
host_status	String	<p>主机状态。</p> <p>uninstall:未安装</p> <p>running:运行</p> <p>offline:离线</p> <p>error:异常</p> <p>plugin error:插件错误</p> <p>installing:安装中</p> <p>install-fail:安装失败</p> <p>upgrading:升级中</p> <p>upgrading-transient:升级中</p> <p>upgrade failed:升级失败</p> <p>upgrade-fail:升级失败</p> <p>uninstalling:卸载中</p> <p>uninstalling-transient:卸载中</p> <p>authentication error:鉴权失败</p> <p>枚举值:</p> <ul style="list-style-type: none"> • uninstall • running • offline • error • plugin error • installing • install-fail • upgrading • upgrading-transient • upgrade failed • upgrade-fail • uninstalling • uninstalling-transient • authentication error
host_type	String	<p>主机类型。linux:linux类型,windows:windows类型</p> <p>枚举值:</p> <ul style="list-style-type: none"> • linux • windows

参数	参数类型	描述
host_version	String	主机版本 最小长度: 1 最大长度: 16
update_time	Long	更新时间 最小值: 13 最大值: 13

状态码: 400

表 6-7 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度: 8 最大长度: 8
error_msg	String	错误描述 最小长度: 1 最大长度: 1000

状态码: 500

表 6-8 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度: 8 最大长度: 8
error_msg	String	错误描述 最小长度: 1 最大长度: 1000

请求示例

查询主机信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
POST https://{endpoint}/v3/{project_id}/lts/host-list
{
```

```
"host_id_list" : [ "713a9f81-574b-45aa-92df-24c4caxxxxxx", "c7085aa9-2142-4ada-9f78-bf81ffxxxxx" ],
"filter" : {
  "host_name_list" : [ "ecs-xxxx", "10.66.16xxx" ],
  "host_ip_list" : [ "192.168xxxx" ],
  "host_status" : "ruxxxx",
  "host_version" : "5.13.xxxx"
}
}
```

响应示例

状态码：200

查询主机信息请求响应成功

```
{
  "result" : [ {
    "host_id" : "dc1dab7e-b045-4e77-bda4-914xxxxxx",
    "host_ip" : "172.16.xxxx",
    "host_name" : "ecs-apmtexxxxxx",
    "host_status" : "xxxxxx",
    "host_type" : "xxxxxx",
    "host_version" : "5.13.xx.x",
    "update_time" : 1637223314526
  } ],
  "total" : 1
}
```

状态码：400

BadRequest 非法请求 建议根据error_msg直接修改该请求。

```
{
  "error_code" : "LTS.1807",
  "error_msg" : "Invalid host id"
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
  "error_code" : "LTS.0010",
  "error_msg" : "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询主机信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;
```

```
import java.util.List;
import java.util.ArrayList;

public class ListHostSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListHostRequest request = new ListHostRequest();
        GetHostListRequestBody body = new GetHostListRequestBody();
        List<String> listFilterHostIpList = new ArrayList<>();
        listFilterHostIpList.add("192.168xxx");
        List<String> listFilterHostNameList = new ArrayList<>();
        listFilterHostNameList.add("ecs-xxx");
        listFilterHostNameList.add("10.66.16xxx");
        GetHostListFilter filterbody = new GetHostListFilter();
        filterbody.withHostNameList(listFilterHostNameList)
            .withHostIpList(listFilterHostIpList)
            .withHostStatus(GetHostListFilter.HostStatusEnum.fromValue("ruxxxx"))
            .withHostVersion("5.13.xxx");
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("713a9f81-574b-45aa-92df-24c4caxxxxx");
        listbodyHostIdList.add("c7085aa9-2142-4ada-9f78-bf81ffxxxx");
        body.withFilter(filterbody);
        body.withHostIdList(listbodyHostIdList);
        request.withBody(body);
        try {
            ListHostResponse response = client.listHost(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

查询主机信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListHostRequest()
        listHostIpListFilter = [
            "192.168xxxx"
        ]
        listHostNameListFilter = [
            "ecs-xxxx",
            "10.66.16xxx"
        ]
        filterbody = GetHostListFilter(
            host_name_list=listHostNameListFilter,
            host_ip_list=listHostIpListFilter,
            host_status="ruxxxx",
            host_version="5.13.xxxx"
        )
        listHostIdListbody = [
            "713a9f81-574b-45aa-92df-24c4caxxxxxx",
            "c7085aa9-2142-4ada-9f78-bf81ffxxxxx"
        ]
        request.body = GetHostListRequestBody(
            filter=filterbody,
            host_id_list=listHostIdListbody
        )
        response = client.list_host(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

查询主机信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListHostRequest{}
var listHostIpListFilter = List<String>{
    "192.168xxx",
}
var listHostNameListFilter = List<String>{
    "ecs-xxx",
    "10.66.16xxx",
}
hostStatusFilter:= model.GetHostListFilterHostStatusEnum().RUXXXX
hostVersionFilter:= "5.13.xxx"
filterbody := &model.GetHostListFilter{
    HostNameList: &listHostNameListFilter,
    HostIpList: &listHostIpListFilter,
    HostStatus: &hostStatusFilter,
    HostVersion: &hostVersionFilter,
}
var listHostIdListbody = List<String>{
    "713a9f81-574b-45aa-92df-24c4caxxxxx",
    "c7085aa9-2142-4ada-9f78-bf81ffxxxx",
}
request.Body = &model.GetHostListRequestBody{
    Filter: filterbody,
    HostIdList: &listHostIdListbody,
}
response, err := client.ListHost(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	查询主机信息请求响应成功
400	BadRequest 非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错

错误码

请参见[错误码](#)。

6.1.2 查询主机组

功能介绍

查询主机组列表

调用方法

请参见[如何调用API](#)。

URI

POST /v3/{project_id}/lts/host-group-list

表 6-9 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：1 最大长度：64

请求参数

表 6-10 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1 最大长度：10000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-11 请求 Body 参数

参数	是否必选	参数类型	描述
host_group_id_list	否	Array of strings	主机组ID 最小长度：36 最大长度：36
filter	是	GetHostGroupListFilter object	主机组过滤参数

表 6-12 GetHostGroupListFilter

参数	是否必选	参数类型	描述
host_group_type	否	String	主机组类型。windows: windows类型, linux: linux类型 枚举值: <ul style="list-style-type: none"> • windows • linux
host_group_name_list	否	Array of strings	主机组名称列表。 最小长度：1 最大长度：64
host_name_list	否	Array of strings	主机名称列表。 最小长度：1 最大长度：128
host_group_tag	否	GetHostGroupListTag object	主机组标签信息

表 6-13 GetHostGroupListTag

参数	是否必选	参数类型	描述
tag_type	否	String	标签类型。AND: 标签过滤的逻辑为与, OR: 标签过滤的逻辑为或 枚举值: <ul style="list-style-type: none"> • AND • OR

参数	是否必选	参数类型	描述
tag_list	否	Array of HostGroupTag objects	主机组标签

表 6-14 HostGroupTag

参数	是否必选	参数类型	描述
key	否	String	标签Key 最小长度：1 最大长度：128
value	否	String	标签Value 最小长度：0 最大长度：255

响应参数

状态码：200

表 6-15 响应 Body 参数

参数	参数类型	描述
result	Array of GetHostGroupInfo objects	主机组列表
total	Long	主机组信息总数量 最小值：0 最大值：1000

表 6-16 GetHostGroupInfo

参数	参数类型	描述
host_group_id	String	主机组ID 最小长度：36 最大长度：36
host_group_name	String	主机组名称 最小长度：1 最大长度：64

参数	参数类型	描述
host_group_type	String	主机组类型。linux: linux类型, windows: windows类型 枚举值: <ul style="list-style-type: none"> • linux • windows
host_id_list	Array of strings	主机ID列表 最小长度: 36 最大长度: 36
host_group_tag	Array of HostGroupTagResBody objects	标签信息.
create_time	Long	创建时间 最小值: 0 最大值: 9999999999999
update_time	Long	更新时间 最小值: 0 最大值: 9999999999999

表 6-17 HostGroupTagResBody

参数	参数类型	描述
key	String	标签Key 最小长度: 1 最大长度: 36
value	String	标签Value 最小长度: 1 最大长度: 36

状态码: 400

表 6-18 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度: 8 最大长度: 8

参数	参数类型	描述
error_msg	String	错误描述 最小长度：1 最大长度：1000

状态码：500

表 6-19 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

请求示例

查询主机组信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
POST https://{endpoint}/v3/{project_id}/lts/host-group-list
{
  "host_group_id_list": [ "bca6d903-3528-42a8-91f4-586722cxxxx" ],
  "filter": {
    "host_group_type": "lixxx",
    "host_group_name_list": [ "wjyTxxx", "xxxest" ],
    "host_name_list": [ "ecs-apmtxxxx删" ],
    "host_group_tag": {
      "tag_type": "AND",
      "tag_list": [ {
        "key": "xxx",
        "value": "xxx"
      } ]
    }
  }
}
```

响应示例

状态码：200

查询主机组列表请求响应成功

```
{
  "result": [ {
    "host_group_id": "598c77aa-c69b-42f0-8cb8-xxxx5b38",
    "host_group_name": "devspoxxxou1",
    "host_group_type": "lxux",
  } ]
}
```

```
"host_id_list" : [ "dc1dab7e-b04xxxx", "xxxxx" ],
"host_group_tag" : [ {
  "key" : "xxx",
  "value" : "xxx"
}, {
  "key" : "xxx",
  "value" : "xxx"
} ],
"create_time" : 1635459410332,
"update_time" : 163560332
}],
"total" : 1
}
```

状态码：400

非法请求 建议根据error_msg直接修改该请求。

```
{
  "error_code" : "LTS.1807",
  "error_msg" : "Invalid host group id"
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code" : "LTS.0010",
  "error_msg" : "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

查询主机组信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListHostGroupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
    }
}
```

```
ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

LtsClient client = LtsClient.newBuilder()
    .withCredential(auth)
    .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
    .build();
ListHostGroupRequest request = new ListHostGroupRequest();
GetHostGroupListRequestBody body = new GetHostGroupListRequestBody();
List<HostGroupTag> listHostGroupTagTagList = new ArrayList<>();
listHostGroupTagTagList.add(
    new HostGroupTag()
        .withKey("xxx")
        .withValue("xxx")
);
GetHostGroupListTag hostGroupTagFilter = new GetHostGroupListTag();
hostGroupTagFilter.withTagType(GetHostGroupListTag.TagTypeEnum.fromValue("AND"))
    .withTagList(listHostGroupTagTagList);
List<String> listFilterHostNameList = new ArrayList<>();
listFilterHostNameList.add("ecs-apmtexxx删");
List<String> listFilterHostGroupNameList = new ArrayList<>();
listFilterHostGroupNameList.add("wjyTxxx");
listFilterHostGroupNameList.add("xxxest");
GetHostGroupListFilter filterbody = new GetHostGroupListFilter();
filterbody.withHostGroupType(GetHostGroupListFilter.HostGroupTypeEnum.fromValue("lixxx"))
    .withHostGroupNameList(listFilterHostGroupNameList)
    .withHostNameList(listFilterHostNameList)
    .withHostGroupTag(hostGroupTagFilter);
List<String> listbodyHostGroupIdList = new ArrayList<>();
listbodyHostGroupIdList.add("bca6d903-3528-42a8-91f4-586722cxxxxx");
body.withFilter(filterbody);
body.withHostGroupIdList(listbodyHostGroupIdList);
request.withBody(body);
try {
    ListHostGroupResponse response = client.listHostGroup(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

查询主机组信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(LtsRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListHostGroupRequest()
    listTagListHostGroupTag = [
        HostGroupTag(
            key="xxx",
            value="xxx"
        )
    ]
    hostGroupTagFilter = GetHostGroupListTag(
        tag_type="AND",
        tag_list=listTagListHostGroupTag
    )
    listHostNameListFilter = [
        "ecs-apmtexxx删"
    ]
    listHostGroupNameListFilter = [
        "wjyTxxx",
        "xxxest"
    ]
    filterbody = GetHostGroupListFilter(
        host_group_type="lixxx",
        host_group_name_list=listHostGroupNameListFilter,
        host_name_list=listHostNameListFilter,
        host_group_tag=hostGroupTagFilter
    )
    listHostGroupIdListbody = [
        "bca6d903-3528-42a8-91f4-586722cxxxxx"
    ]
    request.body = GetHostGroupListRequestBody(
        filter=filterbody,
        host_group_id_list=listHostGroupIdListbody
    )
    response = client.list_host_group(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

查询主机组信息，根据传入的Body体进行过滤。若是body体中无过滤参数，则查询全量数据。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListHostGroupRequest{}
keyTagList:= "xxx"
valueTagList:= "xxx"
var listTagListHostGroupTag = []model.HostGroupTag{
    {
        Key: &keyTagList,
        Value: &valueTagList,
    },
}
tagTypeHostGroupTag:= model.GetGetHostGroupListTagTypeEnum().AND
hostGroupTagFilter := &model.GetHostGroupListTag{
    TagType: &tagTypeHostGroupTag,
    TagList: &listTagListHostGroupTag,
}
var listHostNameListFilter = []string{
    "ecs-apmtexx删",
}
var listHostGroupNameListFilter = []string{
    "wjyTxxx",
    "xxxest",
}
hostGroupTypeFilter:= model.GetGetHostGroupListFilterHostGroupTypeEnum().LIXXX
filterbody := &model.GetHostGroupListFilter{
    HostGroupType: &hostGroupTypeFilter,
    HostGroupNameList: &listHostGroupNameListFilter,
    HostNameList: &listHostNameListFilter,
    HostGroupTag: hostGroupTagFilter,
}
var listHostGroupIdListbody = []string{
    "bca6d903-3528-42a8-91f4-586722cxxxx",
}
request.Body = &model.GetHostGroupListRequestBody{
    Filter: filterbody,
    HostGroupIdList: &listHostGroupIdListbody,
}
response, err := client.ListHostGroup(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	查询主机组列表请求响应成功
400	非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.1.3 创建主机组

功能介绍

创建主机组

调用方法

请参见[如何调用API](#)。

URI

POST /v3/{project_id}/lts/host-group

表 6-20 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：1 最大长度：64

请求参数

表 6-21 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1 最大长度：10000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-22 请求 Body 参数

参数	是否必选	参数类型	描述
host_group_name	是	String	主机组名称 最小长度：1 最大长度：64
host_group_type	是	String	主机组类型。windows：windows类型，linux：linux类型 枚举值： <ul style="list-style-type: none">• linux• windows
host_id_list	否	Array of strings	主机组ID列表。主机类型必须与主机组类型一致 最小长度：36 最大长度：36
host_group_tag	否	Array of HostGroupTag objects	标签信息。最多支持20个标签。KEY不能重复。

表 6-23 HostGroupTag

参数	是否必选	参数类型	描述
key	否	String	标签Key 最小长度：1 最大长度：128
value	否	String	标签Value 最小长度：0 最大长度：255

响应参数

状态码：200

表 6-24 响应 Body 参数

参数	参数类型	描述
host_group_id	String	主机组ID 最小长度：36 最大长度：36
host_group_name	String	主机组名称 最小长度：1 最大长度：64
host_group_type	String	主机组类型。linux：linux类型， windows：windows类型 枚举值： • linux • windows
host_id_list	Array of strings	主机ID列表 最小长度：36 最大长度：36
host_group_tag	Array of HostGroupTagResponseBody objects	标签信息.
create_time	Long	创建时间 最小值：0 最大值：9999999999999

参数	参数类型	描述
update_time	Long	更新时间 最小值：0 最大值：999999999999999

表 6-25 HostGroupTagResBody

参数	参数类型	描述
key	String	标签Key 最小长度：1 最大长度：36
value	String	标签Value 最小长度：1 最大长度：36

状态码：400

表 6-26 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

状态码：500

表 6-27 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8

参数	参数类型	描述
error_msg	String	错误描述 最小长度: 1 最大长度: 1000

请求示例

创建主机组，host_group_name和host_group_type为必填参数

POST https://{endpoint}/v3/{project_id}/lts/host-group

```
{
  "host_group_name": "APIxx3",
  "host_group_type": "linxx",
  "host_id_list": [ "dc1dab7e-b045-4e77xxd1bf7", "713a9fxx2df-24c4ca599def" ],
  "host_group_tag": [ {
    "key": "xxx",
    "value": "xxx"
  }, {
    "key": "xxx1",
    "value": "xxx"
  } ]
}
```

响应示例

状态码: 200

创建主机组请求响应成功

```
{
  "host_group_id": "598c77aa-c69b-42f0-8cb8-983178ad5b38",
  "host_group_name": "devspore_app_dzhou1",
  "host_group_type": "linux",
  "host_id_list": [ "dc1dab7e-b045-4e77-bda4-914d083d1bf7", "xxxxx" ],
  "host_group_tag": [ {
    "key": "xxx",
    "value": "xxx"
  } ],
  "create_time": 1635149410332,
  "update_time": 1635149410332
}
```

状态码: 400

非法请求 建议根据error_msg直接修改该请求。

```
{
  "error_code": "LTS.1807",
  "error_msg": "Invalid host group id"
}
```

状态码: 500

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code": "LTS.0010",
  "error_msg": "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建主机组，host_group_name和host_group_type为必填参数

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateHostGroupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateHostGroupRequest request = new CreateHostGroupRequest();
        CreateHostGroupRequestBody body = new CreateHostGroupRequestBody();
        List<HostGroupTag> listbodyHostGroupTag = new ArrayList<>();
        listbodyHostGroupTag.add(
            new HostGroupTag()
                .withKey("xxx")
                .withValue("xxx")
        );
        listbodyHostGroupTag.add(
            new HostGroupTag()
                .withKey("xxx1")
                .withValue("xxx")
        );
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("dc1dab7e-b045-4e77xxd1bf7");
        listbodyHostIdList.add("713a9fxx2df-24c4ca599def");
        body.withHostGroupTag(listbodyHostGroupTag);
        body.withHostIdList(listbodyHostIdList);
        body.withHostGroupType(CreateHostGroupRequestBody.HostGroupTypeEnum.fromValue("linxx"));
        body.withHostGroupName("APIxx3");
        request.withBody(body);
        try {
            CreateHostGroupResponse response = client.createHostGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

创建主机组，host_group_name和host_group_type为必填参数

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateHostGroupRequest()
        listHostGroupTagbody = [
            HostGroupTag(
                key="xxx",
                value="xxx"
            ),
            HostGroupTag(
                key="xxx1",
                value="xxx"
            )
        ]
        listHostIdListbody = [
            "dc1dab7e-b045-4e77xxd1bf7",
            "713a9fxx2df-24c4ca599def"
        ]
        request.body = CreateHostGroupRequestBody(
            host_group_tag=listHostGroupTagbody,
            host_id_list=listHostIdListbody,
            host_group_type="linxx",
            host_group_name="APIxx3"
        )
        response = client.create_host_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

创建主机组，host_group_name和host_group_type为必填参数

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateHostGroupRequest{}
    keyHostGroupTag := "xxx"
    valueHostGroupTag := "xxx"
    keyHostGroupTag1 := "xxx1"
    valueHostGroupTag1 := "xxx"
    var listHostGroupTagbody = []model.HostGroupTag{
        {
            Key: &keyHostGroupTag,
            Value: &valueHostGroupTag,
        },
        {
            Key: &keyHostGroupTag1,
            Value: &valueHostGroupTag1,
        },
    }
    var listHostIdListbody = []string{
        "dc1dab7e-b045-4e77xxd1bf7",
        "713a9fxx2df-24c4ca599def",
    }
    request.Body = &model.CreateHostGroupRequestBody{
        HostGroupTag: &listHostGroupTagbody,
        HostIdList: &listHostIdListbody,
        HostGroupType: model.GetCreateHostGroupRequestBodyHostGroupTypeEnum().LINXX,
        HostGroupName: "APIxx3",
    }
    response, err := client.CreateHostGroup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	创建主机组请求响应成功
400	非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.1.4 删除主机组

功能介绍

删除主机组

调用方法

请参见[如何调用API](#)。

URI

DELETE /v3/{project_id}/lts/host-group

表 6-28 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：1 最大长度：64

请求参数

表 6-29 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token， 获取方式请参见： 获取用户Token 最小长度：1 最大长度：10000
Content-Type	是	String	该字段填为：application/ json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-30 请求 Body 参数

参数	是否必选	参数类型	描述
host_group_id_list	是	Array of strings	主机组ID列表 最小长度：36 最大长度：36

响应参数

状态码：200

表 6-31 响应 Body 参数

参数	参数类型	描述
result	Array of GetHostGroupInfo objects	主机组详细信息
total	Long	删除主机组数量 最小值：0 最大值：1000

表 6-32 GetHostGroupInfo

参数	参数类型	描述
host_group_id	String	主机组ID 最小长度：36 最大长度：36
host_group_name	String	主机组名称 最小长度：1 最大长度：64
host_group_type	String	主机组类型。linux: linux类型, windows: windows类型 枚举值: <ul style="list-style-type: none">linuxwindows
host_id_list	Array of strings	主机ID列表 最小长度：36 最大长度：36
host_group_tag	Array of HostGroupTagResBody objects	标签信息.
create_time	Long	创建时间 最小值：0 最大值：9999999999999
update_time	Long	更新时间 最小值：0 最大值：9999999999999

表 6-33 HostGroupTagResBody

参数	参数类型	描述
key	String	标签Key 最小长度：1 最大长度：36
value	String	标签Value 最小长度：1 最大长度：36

状态码：400

表 6-34 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

状态码：500

表 6-35 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

请求示例

删除主机组,可同时删除多个主机组。

```
DELETE https://{endpoint}/v3/{project_id}/lts/host-group
/v3/{project_id}/lts/host-group
{"host_group_id_list":["xxx","xxx"]}
```

响应示例

状态码：200

删除主机组列表请求响应成功

```
{
  "result": [{
    "host_group_id": "598c77aa-c69b-42f0-8cb8-xxxx5b38",
    "host_group_name": "devspoxxxou1",
    "host_group_type": "lxxux",
    "host_id_list": ["dc1dab7e-b04xxxx", "xxxxx"],
    "host_group_tag": [{
      "key": "xxx",
      "value": "xxx"
    }, {
      "key": "xxx",
      "value": "xxx"
    }
  ]
}
```

```
    }  
    ],  
    "create_time" : 1635xx9410332,  
    "update_time" : 163xx0332  
  }  
],  
"total" : 1  
}
```

状态码：400

非法请求 建议根据error_msg直接修改该请求。

```
{  
  "error_code" : "LTS.1807",  
  "error_msg" : "Invalid host group id"  
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code" : "LTS.0010",  
  "error_msg" : "Internal Server Error"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

删除主机组,可同时删除多个主机组。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;  
import com.huaweicloud.sdk.lts.v2.*;  
import com.huaweicloud.sdk.lts.v2.model.*;  
  
public class DeleteHostGroupSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        LtsClient client = LtsClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
```

```
        .build();
DeleteHostGroupRequest request = new DeleteHostGroupRequest();
try {
    DeleteHostGroupResponse response = client.deleteHostGroup(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

删除主机组,可同时删除多个主机组。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteHostGroupRequest()
        response = client.delete_host_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

删除主机组,可同时删除多个主机组。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteHostGroupRequest{}
    response, err := client.DeleteHostGroup(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	删除主机组列表请求响应成功
400	非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.1.5 修改主机组

功能介绍

修改主机组

调用方法

请参见[如何调用API](#)。

URI

PUT /v3/{project_id}/lts/host-group

表 6-36 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：1 最大长度：64

请求参数

表 6-37 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1 最大长度：10000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-38 请求 Body 参数

参数	是否必选	参数类型	描述
host_group_id	是	String	主机组ID 最小长度：36 最大长度：36
host_group_name	否	String	主机组名称。不修改主机组名称时不可以传入。 最小长度：1 最大长度：64

参数	是否必选	参数类型	描述
host_id_list	否	Array of strings	主机ID列表。主机类型必须与主机组类型一致 最小长度：36 最大长度：36
host_group_tag	否	Array of HostGroupTag objects	主机组标签。KEY不能重复

表 6-39 HostGroupTag

参数	是否必选	参数类型	描述
key	否	String	标签Key 最小长度：1 最大长度：128
value	否	String	标签Value 最小长度：0 最大长度：255

响应参数

状态码：200

表 6-40 响应 Body 参数

参数	参数类型	描述
host_group_id	String	主机组ID 最小长度：36 最大长度：36
host_group_name	String	主机组名称 最小长度：1 最大长度：64
host_group_type	String	主机组类型。linux：linux类型，windows：windows类型 枚举值： <ul style="list-style-type: none"> linux windows
host_id_list	Array of strings	主机ID列表

参数	参数类型	描述
host_group_tag	Array of HostGroupTag objects	标签信息。最多支持20个标签。
create_time	Long	创建时间 最小值：0 最大值：9999999999999
update_time	Long	更新时间 最小值：0 最大值：9999999999999

表 6-41 HostGroupTag

参数	参数类型	描述
key	String	标签Key 最小长度：1 最大长度：128
value	String	标签Value 最小长度：0 最大长度：255

状态码：400

表 6-42 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

状态码：500

表 6-43 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码 最小长度：8 最大长度：8
error_msg	String	错误描述 最小长度：1 最大长度：1000

请求示例

更新主机组,host_group_id为必填参数。

```
PUT https://{endpoint}/v3/{project_id}/lts/host-group
```

```
{
  "host_group_id": "xxxxxx",
  "host_group_name": "qweqwe",
  "host_id_list": [ "host_id_1", "host_id_2" ],
  "host_group_tag": [ {
    "key": "xxx",
    "value": "xxx"
  } ]
}
```

响应示例

状态码：200

更新主机组请求响应成功

```
{
  "host_group_id": "598c77aa-c69b-42f0-8cb8-983178b38",
  "host_group_name": "devspore_app_dzhou1",
  "host_group_type": "linux",
  "host_id_list": [ "dc1dab7e-b045-4e77-bda4-914d3f7", "xxxxx" ],
  "host_group_tag": [ {
    "key": "xxx",
    "value": "xxx"
  }, {
    "key": "xxx",
    "value": "xxx"
  } ],
  "create_time": 16351494332,
  "update_time": 16351494332
}
```

状态码：400

非法请求 建议根据error_msg直接修改该请求。

```
{
  "error_code": "LTS.1807",
  "error_msg": "Invalid host group id"
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code" : "LTS.0010",
  "error_msg" : "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新主机组,host_group_id为必填参数。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateHostGroupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateHostGroupRequest request = new UpdateHostGroupRequest();
        UpdateHostGroupRequestBody body = new UpdateHostGroupRequestBody();
        List<HostGroupTag> listbodyHostGroupTag = new ArrayList<>();
        listbodyHostGroupTag.add(
            new HostGroupTag()
                .withKey("xxx")
                .withValue("xxx")
        );
        List<String> listbodyHostIdList = new ArrayList<>();
        listbodyHostIdList.add("host_id_1");
        listbodyHostIdList.add("host_id_2");
        body.withHostGroupTag(listbodyHostGroupTag);
        body.withHostIdList(listbodyHostIdList);
        body.withHostGroupName("qweqwe");
        body.withHostGroupId("xxxxxx");
        request.withBody(body);
        try {
```

```
        UpdateHostGroupResponse response = client.updateHostGroup(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

更新主机组,host_group_id为必填参数。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudskcls.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudskcls.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateHostGroupRequest()
        listHostGroupTagbody = [
            HostGroupTag(
                key="xxx",
                value="xxx"
            )
        ]
        listHostIdListbody = [
            "host_id_1",
            "host_id_2"
        ]
        request.body = UpdateHostGroupRequestBody(
            host_group_tag=listHostGroupTagbody,
            host_id_list=listHostIdListbody,
            host_group_name="qweqwe",
            host_group_id="xxxxxx"
        )
        response = client.update_host_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

更新主机组,host_group_id为必填参数。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateHostGroupRequest{}
    keyHostGroupTag := "xxx"
    valueHostGroupTag := "xxx"
    var listHostGroupTagbody = []model.HostGroupTag{
        {
            Key: &keyHostGroupTag,
            Value: &valueHostGroupTag,
        },
    }
    var listHostIdListbody = []string{
        "host_id_1",
        "host_id_2",
    }
    hostGroupNameUpdateHostGroupRequestBody := "qweqwe"
    request.Body = &model.UpdateHostGroupRequestBody{
        HostGroupTag: &listHostGroupTagbody,
        HostIdList: &listHostIdListbody,
        HostGroupName: &hostGroupNameUpdateHostGroupRequestBody,
        HostGroupId: "xxxxx",
    }
    response, err := client.UpdateHostGroup(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	更新主机组请求响应成功
400	非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.2 日志组管理

6.2.1 创建日志组

功能介绍

该接口用于创建一个日志组

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/groups

表 6-44 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32

表 6-45 Query 参数

参数	是否必选	参数类型	描述
enterprise_project_id	否	String	企业项目id

请求参数

表 6-46 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-47 请求 Body 参数

参数	是否必选	参数类型	描述
log_group_name	是	String	需要创建的日志组名称。 配置规则： 名称长度限制为1~64个字符。 名称只能由英文大小写字母、数字、特殊字符“_”“-”“.”组成，且不能以小数点、下划线开头或以小数点结尾。 最小长度：1 最大长度：64
ttl_in_days	是	Integer	日志存储时间（默认为30天）。 最小值：1 最大值：365
tags	否	Array of tagsBody objects	标签字段信息

参数	是否必选	参数类型	描述
log_group_name_alias	否	String	日志组别名

表 6-48 tagsBody

参数	是否必选	参数类型	描述
key	否	String	标签键
value	否	String	标签值

响应参数

状态码：201

表 6-49 响应 Body 参数

参数	参数类型	描述
log_group_id	String	创建的日志组的ID。 最小长度：36 最大长度：36

状态码：400

表 6-50 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-51 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-52 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-53 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：503

表 6-54 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

创建日志组

```
POST https://{endpoint}/v2/{project_id}/groups
```

```
{  
  "log_group_name": "lts-group-01nh",  
  "ttl_in_days": 7  
}
```

响应示例

状态码：201

请求响应成功, 成功创建日志组。

```
{  
  "log_group_id": "b6b9332b-091f-4b22-b810-264318d2d664"  
}
```

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{  
  "error_code": "LTS.0009",  
  "error_msg": "Failed to validate the request body"  
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{  
  "error_code": "LTS.0003",  
  "error_msg": "Invalid token"  
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{  
  "error_code": "LTS.0001",
```

```
"error_msg" : "Invalid projectId"  
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code" : "LTS.0102",  
  "error_msg" : "Failed to create log group"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建日志组

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;  
import com.huaweicloud.sdk.lts.v2.*;  
import com.huaweicloud.sdk.lts.v2.model.*;  
  
public class CreateLogGroupSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        LtsClient client = LtsClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateLogGroupRequest request = new CreateLogGroupRequest();  
        CreateLogGroupParams body = new CreateLogGroupParams();  
        body.withTtlInDays(7);  
        body.withLogGroupName("lts-group-01nh");  
        request.withBody(body);  
        try {  
            CreateLogGroupResponse response = client.createLogGroup(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

创建日志组

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLogGroupRequest()
        request.body = CreateLogGroupParams(
            ttl_in_days=7,
            log_group_name="lts-group-01nh"
        )
        response = client.create_log_group(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建日志组

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateLogGroupRequest{}
request.Body = &model.CreateLogGroupParams{
    TtlInDays: int32(7),
    LogGroupName: "lts-group-01nh",
}
response, err := client.CreateLogGroup(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	请求响应成功, 成功创建日志组。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。
503	ServiceUnavailable。 被请求的服务无效, 服务不可用。

错误码

请参见[错误码](#)。

6.2.2 查询账号下所有日志组

功能介绍

该接口用于查询账号下所有日志组。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/groups

表 6-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32

请求参数

表 6-56 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

响应参数

状态码：200

表 6-57 响应 Body 参数

参数	参数类型	描述
log_groups	Array of LogGroup objects	日志组信息。

表 6-58 LogGroup

参数	参数类型	描述
creation_time	Long	日志组创建时间。
log_group_name	String	日志组名称。 最小长度：1 最大长度：64
log_group_id	String	日志组ID。 最小长度：36 最大长度：36
ttl_in_days	Integer	日志存储时间（天）。 最小值：1 最大值：365
tag	Map<String,String>	日志组标签。

状态码：401

表 6-59 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-60 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-61 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

查询账号下的所有日志组

```
GET https://{endpoint}/v2/{project_id}/groups
/v2/{project_id}/groups
```

响应示例

状态码：200

请求响应成功。

```
{
  "log_groups" : [ {
    "creation_time" : 1630547141853,
    "log_group_name" : "lts-group-01nh",
    "log_group_id" : "b6b9332b-091f-4b22-b810-264318d2d664",
    "ttl_in_days" : 7
  } ]
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{  
  "error_code": "LTS.0003",  
  "error_msg": "Invalid token"  
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{  
  "error_code": "LTS.0001",  
  "error_msg": "Invalid projectId"  
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code": "LTS.0010",  
  "error_msg": "The system encountered an internal error"  
}
```

状态码

状态码	描述
200	请求响应成功。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.2.3 删除日志组

功能介绍

该接口用于删除指定日志组。当日志组中的日志流配置了日志转储，需要取消日志转储后才可删除。

调用方法

请参见[如何调用API](#)。

URI

DELETE /v2/{project_id}/groups/{log_group_id}

表 6-62 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 。 最小长度：36 最大长度：36

请求参数

表 6-63 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

响应参数

状态码：400

表 6-64 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-65 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-66 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-67 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

删除日志组

```
DELETE https://{endpoint}/v2/{project_id}/groups/{log_group_id}
/v2/{project_id}/groups/{log_group_id}
```

响应示例

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0201",
  "error_msg": "The log group is not existed"
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
  "error_code": "LTS.0003",
  "error_msg": "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0001",
  "error_msg": "Invalid projectId"
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code": "LTS.0103",
```

```
"error_msg": "Failed to delete log group"  
}
```

状态码

状态码	描述
204	请求响应成功, 成功删除日志组。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到, 但是服务内部出错。

错误码

请参见[错误码](#)。

6.2.4 修改日志组

功能介绍

该接口用于修改指定日志组下的日志存储时长。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/groups/{log_group_id}

表 6-68 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 获取方式请参见: 获取项目ID, 获取账号ID, 日志组ID、日志流ID 最小长度: 36 最大长度: 36

参数	是否必选	参数类型	描述
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 。 最小长度：36 最大长度：36

请求参数

表 6-69 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-70 请求 Body 参数

参数	是否必选	参数类型	描述
ttl_in_days	是	Integer	日志存储时间（天）。 最小值：1 最大值：365 缺省值：7
tags	否	Array of tagsBody objects	标签字段信息

表 6-71 tagsBody

参数	是否必选	参数类型	描述
key	否	String	标签键
value	否	String	标签值

响应参数

状态码：200

表 6-72 响应 Body 参数

参数	参数类型	描述
creation_time	Long	创建该日志组的时间，毫秒级。
log_group_name	String	日志组的名称。 最小长度：1 最大长度：64
log_group_id	String	日志组ID。 最小长度：36 最大长度：36
ttl_in_days	Integer	日志存储时间。取值范围为 [1, 365] 最小值：1 最大值：365 枚举值： <ul style="list-style-type: none">7

状态码：400

表 6-73 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： <ul style="list-style-type: none">LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： <ul style="list-style-type: none">Invalid projectId

状态码：401

表 6-74 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-75 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-76 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

修改日志组

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}
```

```
{  
  "ttl_in_days" : 8  
}
```

响应示例

状态码：200

请求响应成功, 成功修改日志组。

```
{  
  "creation_time" : 156541515155541,  
  "log_group_name" : "string",  
  "log_group_id" : "string",  
  "ttl_in_days" : 8  
}
```

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{  
  "error_code" : "LTS.0009",  
  "error_msg" : "Failed to validate the request body"  
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{  
  "error_code" : "LTS.0414",  
  "error_msg" : "Invalid token"  
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{  
  "error_code" : "LTS.0001",  
  "error_msg" : "Invalid projectId"  
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code" : "LTS.0102",  
  "error_msg" : "Failed to update log group"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改日志组

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateLogGroupSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateLogGroupRequest request = new UpdateLogGroupRequest();
        request.withLogGroupId("{log_group_id}");
        UpdateLogGroupParams body = new UpdateLogGroupParams();
        body.withTtlInDays(8);
        request.withBody(body);
        try {
            UpdateLogGroupResponse response = client.updateLogGroup(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

修改日志组

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

```
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
# In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(LtsRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateLogGroupRequest()
    request.log_group_id = "{log_group_id}"
    request.body = UpdateLogGroupParams(
        ttl_in_days=8
    )
    response = client.update_log_group(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

修改日志组

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateLogGroupRequest{}
    request.LogGroupId = "{log_group_id}"
    request.Body = &model.UpdateLogGroupParams{
        TtlInDays: int32(8),
```

```
}  
response, err := client.UpdateLogGroup(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求响应成功, 成功修改日志组。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.3 日志流管理

6.3.1 创建日志流

功能介绍

该接口用于创建某个指定日志组下的日志流

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/groups/{log_group_id}/streams

表 6-77 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 。 最小长度：36 最大长度：36

请求参数

表 6-78 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-79 请求 Body 参数

参数	是否必选	参数类型	描述
log_stream_name	是	String	需要创建的日志流名称。 最小长度：1 最大长度：64
ttl_in_days	否	Integer	日志存储时间 最小值：1 最大值：365

参数	是否必选	参数类型	描述
tags	否	Array of tagsBody objects	标签字段信息
log_stream_name_alias	否	String	日志流名称别名 最小长度：1 最大长度：64
enterprise_project_name	否	String	企业项目名称。默认default企业项目，如果企业项目不存在，创建日志流默认default企业项目，可以在企业项目管理页面查看全部企业项目。 最小长度：1 最大长度：255

表 6-80 tagsBody

参数	是否必选	参数类型	描述
key	否	String	标签键
value	否	String	标签值

响应参数

状态码：201

表 6-81 响应 Body 参数

参数	参数类型	描述
log_stream_id	String	创建的日志流的Id。 最小长度：36 最大长度：36

状态码：400

表 6-82 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-83 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-84 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-85 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

创建日志流

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams
{
  "log_stream_name": "lts-stream-02kh"
}
```

响应示例

状态码：201

请求响应成功, 成功创建日志流。

```
{
  "log_stream_id": "c54dbc58-0fd8-48ed-b007-6d54981427a7"
}
```

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0009",
  "error_msg": "Failed to validate the request body"
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
  "error_code": "LTS.0003",
  "error_msg": "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0001",
  "error_msg": "Invalid projectId"
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code" : "LTS.0202",
  "error_msg" : "Failed to create Log stream"
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建日志流

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class CreateLogStreamSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateLogStreamRequest request = new CreateLogStreamRequest();
        request.withLogGroupId("{log_group_id}");
        CreateLogStreamParams body = new CreateLogStreamParams();
        body.withLogStreamName("lts-stream-02kh");
        request.withBody(body);
        try {
            CreateLogStreamResponse response = client.createLogStream(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
        }
    }
}
```

```
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

创建日志流

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLogStreamRequest()
        request.log_group_id = "{log_group_id}"
        request.body = CreateLogStreamParams(
            log_stream_name="lts-stream-02kh"
        )
        response = client.create_log_stream(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建日志流

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateLogStreamRequest{}
request.LogGroupId = "{log_group_id}"
request.Body = &model.CreateLogStreamParams{
    LogStreamName: "lts-stream-02kh",
}
response, err := client.CreateLogStream(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	请求响应成功, 成功创建日志流。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.3.2 查询指定日志组下的所有日志流

功能介绍

该接口用于查询指定日志组下的所有日志流信息。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/groups/{log_group_id}/streams

表 6-86 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 。 最小长度：36 最大长度：36

请求参数

表 6-87 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

响应参数

状态码：200

表 6-88 响应 Body 参数

参数	参数类型	描述
log_streams	Array of LogStreamResBody objects	日志流

表 6-89 LogStreamResBody

参数	参数类型	描述
creation_time	Long	创建时间 最小值：1577808000000 最大值：4102416000000
log_stream_id	String	日志流ID 最小长度：36 最大长度：36
log_stream_name	String	日志流名称 最小长度：1 最大长度：64
log_stream_name_alias	String	日志流别名
tag	Map<String,String>	日志流所属标签
filter_count	Integer	过滤器个数 最小值：0 最大值：5
is_favorite	Boolean	是否收藏日志流。
whether_log_store	Boolean	是否日志存储
hot_cold_separation	Boolean	是否冷存储
auth_web_tracking	Boolean	匿名写入开关
ttl_in_days	Integer	存储时间
hot_storage_days	Integer	标准存储时间

状态码：401**表 6-90** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403**表 6-91** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500**表 6-92** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

查询指定日志组下的所有日志流

```
GET https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams  
/v2/{project_id}/groups/{log_group_id}/streams
```

响应示例

状态码：200

请求响应成功。

```
{  
  "log_streams": [ {  
    "log_stream_name_alias": "3",  
    "creation_time": 1723102152593,  
    "auth_web_tracking": false,  
    "log_stream_name": "3",  
    "is_favorite": false,  
    "hot_storage_days": -1,  
    "tag": {  
      "_sys_enterprise_project_id": "0"  
    },  
    "hot_cold_separation": false,  
    "filter_count": 0,  
    "log_stream_id": "667293b4-7f6d-4efb-9b30-xxxxxxx",  
    "ttl_in_days": -1,  
    "whether_log_storage": true  
  }, {  
    "log_stream_name_alias": "5",  
    "creation_time": 1723102172632,  
    "auth_web_tracking": false,  
    "log_stream_name": "5",  
    "is_favorite": false,  
    "hot_storage_days": -1,  
    "tag": {  
      "_sys_enterprise_project_id": "0"  
    },  
    "hot_cold_separation": false,  
    "filter_count": 0,  
    "log_stream_id": "fb08a859-69ed-4bbe-9ba0-xxxxxxx",  
    "ttl_in_days": -1,  
    "whether_log_storage": true  
  } ]  
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{  
  "error_code": "LTS.0003",  
  "error_msg": "Invalid token"  
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。

```
{  
  "error_code": "LTS.0001",  
  "error_msg": "Invalid projectId"  
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code": "LTS.0010",
  "error_msg": "The system encountered an internal error"
}
```

状态码

状态码	描述
200	请求响应成功。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到, 但是服务内部出错。

错误码

请参见[错误码](#)。

6.3.3 查询日志流信息

功能介绍

该接口用于查询LTS日志流信息。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/log-streams

表 6-93 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 获取方式请参见: 获取项目ID, 获取账号ID, 日志组ID、日志流ID 最小长度: 32 最大长度: 32

表 6-94 Query 参数

参数	是否必选	参数类型	描述
log_group_name	否	String	日志组名称 最小长度：1 最大长度：64
log_stream_name	否	String	日志流名称 最小长度：1 最大长度：64

请求参数

表 6-95 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token， 获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/ json;charset=UTF-8。 最小长度：30 最大长度：30

响应参数

状态码：200

表 6-96 响应 Body 参数

参数	参数类型	描述
log_streams	Array of LogStreamNolsFavorite objects	日志组数组。

表 6-97 LogStreamNolsFavorite

参数	参数类型	描述
creation_time	Long	日志流创建时间 最小值: 1577808000000 最大值: 4102416000000
log_stream_id	String	日志流ID 最小长度: 36 最大长度: 36
log_stream_name	String	日志流名称 最小长度: 1 最大长度: 64
log_stream_name_alias	String	日志流别名
tag	Map<String,String>	日志流所属标签
filter_count	Integer	过滤器个数 最小值: 0 最大值: 5
hot_cold_separation	String	冷热存储开关
whether_log_storage	String	是否日志存储
auth_web_tracking	String	匿名写入开关
ttl_in_days	String	日志存储天数
hot_storage_days	String	标准存储天数

状态码: 400

表 6-98 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 最小长度: 8 最大长度: 8
error_msg	String	调用失败响应信息描述。

状态码：500**表 6-99** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 最小长度：8 最大长度：8
error_msg	String	调用失败响应信息描述。

请求示例

若不传参数则查询所有日志流信息。若根据log_group_name, log_stream_name这2种不同的参数, 则查询对应的日志流信息。

```
GET https://{endpoint}/v2/{project_id}/log-streams
/v2/{project_id}/log-streams /v2/{project_id}/log-streams?log_group_name=lbs-group-txxxx
/v2/{project_id}/log-streams?log_stream_name=lbs-xunjian-topic-xxxx
/v2/{project_id}/log-streams?log_stream_name=lbs-xunjian-topic-xxxx&log_group_name=lbs-group-xxx
```

响应示例

状态码：200

查询日志流信息请求响应成功。

```
{
  "log_streams": [ {
    "log_stream_name_alias": "lbs-topic-xxxx",
    "creation_time": 1698915657847,
    "auth_web_tracking": false,
    "log_stream_name": "lbs-topic-xxxx",
    "is_favorite": false,
    "hot_storage_days": -1,
    "tag": {
      "_sys_enterprise_project_id": "xxxxxxx"
    },
    "hot_cold_separation": false,
    "filter_count": 0,
    "log_stream_id": "xxxxxxx",
    "ttl_in_days": 7,
    "whether_log_storage": true
  }, {
    "log_stream_name_alias": "lbs-topic-xxxx",
    "creation_time": 1698915650021,
    "auth_web_tracking": false,
    "log_stream_name": "lbs-topic-xxxx",
    "is_favorite": false,
    "hot_storage_days": -1,
    "tag": {
      "_sys_enterprise_project_id": "xxxxxxx"
    },
    "hot_cold_separation": false,
    "filter_count": 0,
    "log_stream_id": "xxxxxxxxxxxxxxxxxxxx",
    "ttl_in_days": 7,
    "whether_log_storage": true
  }
  ]
}
```

状态码：400

BadRequest 非法请求 建议根据error_msg直接修改该请求。

```
{  
  "error_code": "LTS.0205",  
  "error_msg": "The log stream name has been existed"  
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code": "LTS.0010",  
  "error_msg": "The system encountered an internal error"  
}
```

状态码

状态码	描述
200	查询日志流信息请求响应成功。
400	BadRequest 非法请求 建议根据error_msg直接修改该请求。
500	表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.3.4 删除日志流

功能介绍

该接口用于删除指定日志组下的指定日志流。当该日志流配置了日志转储，需要取消日志转储后才可删除。

调用方法

请参见[如何调用API](#)。

URI

DELETE /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}

表 6-100 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36
log_stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36

请求参数

表 6-101 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

响应参数

状态码：400

表 6-102 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-103 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-104 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-105 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： <ul style="list-style-type: none">LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： <ul style="list-style-type: none">Invalid projectId

请求示例

删除日志流

```
DELETE https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}
/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}
```

响应示例

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0208",
  "error_msg": "The log stream does not existed"
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
  "error_code": "LTS.0003",
  "error_msg": "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
  "error_code": "LTS.0001",
  "error_msg": "Invalid projectId"
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code": "LTS.0203",
```

```
"error_msg": "Failed to delete Log stream"  
}
```

状态码

状态码	描述
204	请求响应成功, 成功删除日志流。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。
401	AuthFailed。鉴权失败, 请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到, 但是服务内部出错。

错误码

请参见[错误码](#)。

6.3.5 创建日志流索引

功能介绍

该接口用于向指定日志流创建索引。

调用方法

请参见[如何调用API](#)。

URI

POST /v1.0/{project_id}/groups/{group_id}/stream/{stream_id}/index/config

表 6-106 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID, 获取方式请参见: 获取项目ID, 获取账号ID, 日志组ID、日志流ID 最小长度: 36 最大长度: 36

参数	是否必选	参数类型	描述
group_id	是	String	日志组ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36
stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36

请求参数

表 6-107 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token
Content-Type	是	String	该字段填为：application/json;charset=UTF-8

表 6-108 请求 Body 参数

参数	是否必选	参数类型	描述
logStreamId	是	String	日志流ID
fullTextIndex	是	LTSFullTextIndexInfo object	全文索引配置
fields	否	Array of LTSFieldsInfo objects	字段索引配置

表 6-109 LTSFullTextIndexInfo

参数	是否必选	参数类型	描述
enable	是	Boolean	是否开启全文索引

参数	是否必选	参数类型	描述
caseSensitive	是	Boolean	是否大小写敏感
includeChinese	是	Boolean	是否包含中文
tokenizer	是	String	自定义分词符 最小长度： 0 最大长度： 128
ascii	否	Array of strings	特殊分词符 最小长度： 1 最大长度： 3

表 6-110 LTSFieldsInfo

参数	是否必选	参数类型	描述
fieldType	是	String	字段类型 枚举值： <ul style="list-style-type: none"> • string • long • float • json
fieldName	是	String	字段名称 最小长度： 1 最大长度： 256
caseSensitive	否	Boolean	是否大小写敏感 缺省值： false
includeChinese	否	Boolean	是否包含中文 缺省值： false
tokenizer	是	String	分词符 最小长度： 0 最大长度： 128
quickAnalysis	否	Boolean	是否快速分析 缺省值： false
ascii	否	Array of strings	特殊分词符
ltsSubFieldsInfoList	否	Array of LTSSubFieldsInfo objects	json字段信息

表 6-111 LTSSubFieldsInfo

参数	是否必选	参数类型	描述
fieldType	是	String	字段类型 枚举值： <ul style="list-style-type: none"> • string • long • float
fieldName	是	String	字段名称 最小长度：1 最大长度：256
caseSensitive	否	Boolean	是否大小写敏感 缺省值：false
includeChinese	否	Boolean	是否包含中文 缺省值：false
tokenizer	否	String	分词符 最小长度：0 最大长度：128
quickAnalysis	否	Boolean	是否快速分析 缺省值：false
ascii	否	Array of strings	特殊分词符

响应参数

状态码：200

表 6-112 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码
errorMessage	String	错误信息描述
result	String	结果
isQueryComplete	Boolean	是否查询完成

状态码：400

表 6-113 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

状态码：401

表 6-114 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

状态码：500

表 6-115 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

请求示例

该接口用于向指定日志流创建索引。

```
POST https://{endpoint}/v1.0/{project_id}/groups/{group_id}/stream/{stream_id}/index/config
```

```
{
  "logStreamId" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
  "fullTextIndex" : {
    "enable" : true,
    "caseSensitive" : false,
    "includeChinese" : true,
    "tokenizer" : ", \"';=()[]{}@&<>:/:\\n\\t\\r"
  },
  "fields" : [ {
    "fieldType" : "long",
    "fieldName" : "fieldName1",
    "caseSensitive" : false,
    "includeChinese" : true,
    "tokenizer" : "",
    "quickAnalysis" : true
  }, {
    "fieldType" : "string",
    "fieldName" : "fieldName2",
    "caseSensitive" : false,
    "includeChinese" : true,
    "tokenizer" : "",
    "quickAnalysis" : true
  } ]
}
```

响应示例

状态码：200

添加索引配置成功

```
{
  "errorCode" : "SVCSTG.ALS.200200",
  "errorMessage" : "add or update indexConfig successfully",
  "isQueryComplete" : true,
  "result" : "493d04ce-5130-4b07-88be-c5ae3b50bccb"
}
```

状态码：400

请求参数错误，请根据响应体内容进行修改

```
{
  "errorCode" : "SVCSTG.ALS.200201",
  "errorMessage" : "IndexConfigInfo check failed.",
  "isQueryComplete" : true
}
```

状态码：401

认证鉴权失败，排查认证信息是否过期

```
{
  "error_msg" : "Incorrect IAM authentication information: decrypt token fail",
  "error_code" : "APIGW.0301",
  "request_id" : "b16cc9d789f34cd5196d8df065341788"
}
```

状态码：500

表示服务后端已经接收到请求，但服务内部处理错误

```
{
  "highLightFields" : { },
  "errorCode" : "LTS.0203",
  "errorMessage" : "Internal Server Error",
  "result" : null,
  "encoded_authorization_message" : null,
}
```

```
"isQueryComplete" : true  
}
```

SDK 代码示例

SDK代码示例如下。

Java

该接口用于向指定日志流创建索引。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;  
import com.huaweicloud.sdk.lts.v2.*;  
import com.huaweicloud.sdk.lts.v2.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class CreateLogStreamIndexSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        LtsClient client = LtsClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateLogStreamIndexRequest request = new CreateLogStreamIndexRequest();  
        request.withGroupId("{group_id}");  
        request.withStreamId("{stream_id}");  
        LTSIndexConfigInfo body = new LTSIndexConfigInfo();  
        List<LTSFieldsInfo> listbodyFields = new ArrayList<>();  
        listbodyFields.add(  
            new LTSFieldsInfo()  
                .withFieldType(LTSFieldsInfo.FieldTypeEnum.fromValue("long"))  
                .withFieldName("fieldName1")  
                .withCaseSensitive(false)  
                .withIncludeChinese(true)  
                .withTokenizer("")  
                .withQuickAnalysis(true)  
        );  
        listbodyFields.add(  
            new LTSFieldsInfo()  
                .withFieldType(LTSFieldsInfo.FieldTypeEnum.fromValue("string"))  
                .withFieldName("fieldName2")  
                .withCaseSensitive(false)  
                .withIncludeChinese(true)  
                .withTokenizer("")  
        );  
    }  
}
```

```
        .withQuickAnalysis(true)
    );
    LTSFullTextIndexInfo fullTextIndexbody = new LTSFullTextIndexInfo();
    fullTextIndexbody.setEnabled(true)
        .withCaseSensitive(false)
        .withIncludeChinese(true)
        .withTokenizer("",";=()[]{}@&<>/:\\n\\t\\r");
    body.withLogStreamId("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx");
    body.withFields(listbodyFields);
    body.withFullTextIndex(fullTextIndexbody);
    request.withBody(body);
    try {
        CreateLogStreamIndexResponse response = client.createLogStreamIndex(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

该接口用于向指定日志流创建索引。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLogStreamIndexRequest()
        request.group_id = "{group_id}"
        request.stream_id = "{stream_id}"
        listFieldsbody = [
            LTSFieldsInfo(
                field_type="long",
                field_name="fieldName1",
                case_sensitive=False,
                include_chinese=True,
                tokenizer="",
                quick_analysis=True
            )
        ]
```

```
    ),
    LTSFieldsInfo(
        field_type="string",
        field_name="fieldName2",
        case_sensitive=False,
        include_chinese=True,
        tokenizer="",
        quick_analysis=True
    )
]
fullTextIndexbody = LTSFullTextIndexInfo(
    enable=True,
    case_sensitive=False,
    include_chinese=True,
    tokenizer="";=()[]{}@&<>/:\\n\\t\\r"
)
request.body = LTSIndexConfigInfo(
    log_stream_id="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    fields=listFieldsbody,
    full_text_index=fullTextIndexbody
)
response = client.create_log_stream_index(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

该接口用于向指定日志流创建索引。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateLogStreamIndexRequest{}
    request.GroupId = "{group_id}"
    request.StreamId = "{stream_id}"
    caseSensitiveFields:= false
```

```
includeChineseFields:= true
quickAnalysisFields:= true
caseSensitiveFields1:= false
includeChineseFields1:= true
quickAnalysisFields1:= true
var listFieldsbody = []model.LtsFieldsInfo{
    {
        FieldType: model.GetLtsFieldsInfoFieldTypeEnum().LONG,
        FieldName: "fieldName1",
        CaseSensitive: &caseSensitiveFields,
        IncludeChinese: &includeChineseFields,
        Tokenizer: "",
        QuickAnalysis: &quickAnalysisFields,
    },
    {
        FieldType: model.GetLtsFieldsInfoFieldTypeEnum().STRING,
        FieldName: "fieldName2",
        CaseSensitive: &caseSensitiveFields1,
        IncludeChinese: &includeChineseFields1,
        Tokenizer: "",
        QuickAnalysis: &quickAnalysisFields1,
    },
}
fullTextIndexbody := &model.LtsFullTextIndexInfo{
    Enable: true,
    CaseSensitive: false,
    IncludeChinese: true,
    Tokenizer: ", ";=() []{}@&<>/:\n\t\r",
}
request.Body = &model.LtsIndexConfigInfo{
    LogStreamId: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    Fields: &listFieldsbody,
    FullTextIndex: fullTextIndexbody,
}
response, err := client.CreateLogStreamIndex(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	添加索引配置成功
400	请求参数错误，请根据响应体内容进行修改
401	认证鉴权失败，排查认证信息是否过期
500	表示服务后端已经接收到请求，但服务内部处理错误

错误码

请参见[错误码](#)。

6.3.6 修改日志流

功能介绍

该接口用于修改指定日志流下的日志存储时长。

调用方法

请参见[如何调用API](#)。

URI

PUT /v2/{project_id}/groups/{log_group_id}/streams-ttl/{log_stream_id}

表 6-116 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36
log_stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36

请求参数

表 6-117 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000

参数	是否必选	参数类型	描述
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-118 请求 Body 参数

参数	是否必选	参数类型	描述
ttl_in_days	是	Integer	日志存储时间（天）。 最小值：1 最大值：365
tags	否	Array of tagsBody objects	标签字段信息

表 6-119 tagsBody

参数	是否必选	参数类型	描述
key	否	String	标签键
value	否	String	标签值

响应参数

状态码：200

表 6-120 响应 Body 参数

参数	参数类型	描述
creation_time	Long	创建该日志流的时间
log_topic_name	String	日志流的名称。 最小长度：1 最大长度：64
log_topic_id	String	日志流ID。 最小长度：36 最大长度：36
ttl_in_days	Integer	日志存储时间（天）。

状态码：400

表 6-121 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-122 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-123 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-124 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： <ul style="list-style-type: none">• LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： <ul style="list-style-type: none">• Invalid projectId

请求示例

修改日志流

```
PUT https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams-ttl/{log_stream_id}
{
  "ttl_in_days" : 8
}
```

响应示例

状态码：200

请求响应成功，成功修改日志流。

```
{
  "creation_time" : 1629947408497,
  "log_topic_name" : "string",
  "log_topic_id" : "string",
  "ttl_in_days" : 8
}
```

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
  "error_code" : "LTS.0009",
  "error_msg" : "Failed to validate the request body"
}
```

状态码：401

AuthFailed。鉴权失败，请确认token后再次请求。

```
{
  "error_code" : "LTS.0414",
  "error_msg" : "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
  "error_code" : "LTS.0001",
```

```
"error_msg" : "Invalid projectId"  
}
```

状态码：500

InternalServerError。表明服务端能被请求访问到，但是服务内部出错。

```
{  
  "error_code" : "LTS.0204",  
  "error_msg" : "Failed to update log stream"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改日志流

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;  
import com.huaweicloud.sdk.lts.v2.*;  
import com.huaweicloud.sdk.lts.v2.model.*;  
  
public class UpdateLogStreamSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        LtsClient client = LtsClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateLogStreamRequest request = new UpdateLogStreamRequest();  
        request.withLogGroupID("{log_group_id}");  
        request.withLogStreamID("{log_stream_id}");  
        UpdateLogStreamParams body = new UpdateLogStreamParams();  
        body.withTtlInDays(8);  
        request.withBody(body);  
        try {  
            UpdateLogStreamResponse response = client.updateLogStream(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
    }
}
}
```

Python

修改日志流

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateLogStreamRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        request.body = UpdateLogStreamParams(
            ttl_in_days=8
        )
        response = client.update_log_stream(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

修改日志流

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateLogStreamRequest{
    request.LogGroupId = "{log_group_id}"
    request.LogStreamId = "{log_stream_id}"
    request.Body = &model.UpdateLogStreamParams{
        TtlInDays: int32(8),
    }
}
response, err := client.UpdateLogStream(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求响应成功，成功修改日志流。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败，请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。表明服务端能被请求访问到，但是服务内部出错。
503	ServiceUnavailable。被请求的服务无效，服务不可用。

错误码

请参见[错误码](#)。

6.4 日志管理

6.4.1 统计 top n 的日志组或日志流流量

功能介绍

统计top n的日志组或日志流流量

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/lts/topn-traffic-statistics

表 6-125 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32

请求参数

表 6-126 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-127 请求 Body 参数

参数	是否必选	参数类型	描述
end_time	是	Long	结束时间时间戳，毫秒时间
is_desc	是	Boolean	是否降序 true / false
resource_type	是	String	资源类型，log_group / log_stream / tenant log_group: 日志组 log_stream: 日志流 tenant: 租户
sort_by	是	String	排序依据，index/write/storage/basicTransfer/seniorTransfer/coldStorage，必须是search_list中存在的数据 index: 索引 write: 读写 storage: 存储 basicTransfer: 基础转储 seniorTransfer: 高级转储 coldStorage: 冷存储量
start_time	是	Long	开始时间时间戳，毫秒时间，最多支持30天范围内的查询
topn	是	Integer	查询前多少数据，范围1~100
filter	是	Map<String,String>	过滤条件 { "log_group_id": "xxxxxx" }过滤器，为一个map结构，键为过滤属性，值为属性值，不支持模糊匹配
search_list	是	Array of strings	查询数据类型，字符串数组可多种搭配，只能在index/write/storage/basicTransfer/seniorTransfer/coldStorage中选填 index: 索引 write: 读写 storage: 存储 basicTransfer: 基础转储 seniorTransfer: 高级转储 coldStorage: 冷存储量

响应参数

状态码：200

表 6-128 响应 Body 参数

参数	参数类型	描述
results	Array of ResultsTopnBody objects	响应结果

表 6-129 ResultsTopnBody

参数	参数类型	描述
index_traffic	Double	索引流量, byte, 查询数据类型中包含 index 时返回
storage	Double	存储量, byte, 查询数据类型中包含 storage 时返回
write_traffic	Double	写入流量, byte, 查询数据类型中包含 write 时返回
log_group_id	String	日志组 id, 资源类型为日志组时返回
log_group_name	String	日志组名称, 资源类型为日志组时返回
log_stream_id	String	日志流 id, 资源类型为日志流时返回
log_stream_name	String	日志流名称, 资源类型为日志流时返回
basic_transfer	Double	基础转储流量, byte, 查询数据类型中包含 basicTransfer 时返回
senior_transfer	Double	基础转储流量, byte, 查询数据类型中包含 seniorTransfer 时返回
is_agency_transfer	Boolean	不是委托转储, true, 是委托转储, 则前端资源统计展示的流不能跳 缺省值: false
cold_storage	Double	冷存储量

状态码：400

表 6-130 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

状态码：500

表 6-131 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

请求示例

统计top n的日志组或日志流流量

```
/v2/2a473356cca5487f8373be891bffc1cf/lts/topn-traffic-statistics
{
  "sort_by": "storage",
  "is_desc": true,
  "resource_type": "log_stream",
  "filter": { },
  "start_time": 1668668183969,
  "end_time": 1669272983969,
  "search_list": [ "index", "write", "storage" ],
  "topn": 100
}
```

响应示例

状态码：200

查询资源成功

```
{
  "results": [ {
    "index_traffic": 0,
    "log_stream_id": "6fd93d47-7630-4284-a622-311d0082f6bb",
```

```
"log_stream_name" : "cmdb-cce-cluster",
"storage" : 59810657587,
"write_traffic" : 0
}, {
  "index_traffic" : 0,
  "log_stream_id" : "504ec3dd-ac28-4783-babb-22a49f36afe3",
  "log_stream_name" : "CMSkaifatest",
  "storage" : 20033606015,
  "write_traffic" : 0
}, {
  "index_traffic" : 6825703991,
  "log_stream_id" : "a14dacb0-5a13-43a8-89a3-ea5424d95133",
  "log_stream_name" : "ELB",
  "storage" : 15659303771,
  "write_traffic" : 1.3651407982E9
}, {
  "index_traffic" : 302172889,
  "log_stream_id" : "25fe7494-7395-438e-8340-647613673ffa",
  "log_stream_name" : "LTStest-916-statefulset",
  "storage" : 316552589,
  "write_traffic" : 6.04345778E7
}, {
  "index_traffic" : 0,
  "log_stream_id" : "956586fc-b828-44be-8672-0a323962a8fa",
  "log_stream_name" : "mongodb_slow",
  "storage" : 0,
  "write_traffic" : 0
}
}]
}
```

状态码：400

BadRequest 非法请求建议根据error_msg直接修改该请求

```
{
  "errorCode" : "LTS.0208",
  "errorMessage" : "The log stream does not existed"
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
  "errorCode" : "LTS.0203",
  "errorMessage" : "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

统计top n的日志组或日志流流量

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;
```

```
import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class ListTopnTrafficStatisticsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();

        ListTopnTrafficStatisticsRequest request = new ListTopnTrafficStatisticsRequest();
        TopnRequestBody body = new TopnRequestBody();
        List<String> listbodySearchList = new ArrayList<>();
        listbodySearchList.add("index");
        listbodySearchList.add("write");
        listbodySearchList.add("storage");
        body.withSearchList(listbodySearchList);
        body.withTopn(100);
        body.withStartTime(1668668183969L);
        body.withSortBy("storage");
        body.withResourceType("log_stream");
        body.withIsDesc(true);
        body.withEndTime(1669272983969L);
        request.withBody(body);
        try {
            ListTopnTrafficStatisticsResponse response = client.listTopnTrafficStatistics(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

统计top n的日志组或日志流流量

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTopnTrafficStatisticsRequest()
        listSearchListbody = [
            "index",
            "write",
            "storage"
        ]
        request.body = TopnReqstBody(
            search_list=listSearchListbody,
            topn=100,
            start_time=1668668183969,
            sort_by="storage",
            resource_type="log_stream",
            is_desc=True,
            end_time=1669272983969
        )
        response = client.list_topn_traffic_statistics(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

统计top n的日志组或日志流流量

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
```

```
Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListTopnTrafficStatisticsRequest{}
var listSearchListbody = []string{
    "index",
    "write",
    "storage",
}
request.Body = &model.TopnRequestBody{
    SearchList: listSearchListbody,
    Topn: int32(100),
    StartTime: int64(1668668183969),
    SortBy: "storage",
    ResourceType: "log_stream",
    IsDesc: true,
    EndTime: int64(1669272983969),
}
response, err := client.ListTopnTrafficStatistics(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	查询资源成功
400	BadRequest 非法请求建议根据error_msg直接修改该请求
500	表明服务端能被请求访问到，但是服务内部出错

错误码

请参见[错误码](#)。

6.4.2 按时间段统计查询资源

功能介绍

按时间段统计查询资源。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/lts/timeline-traffic-statistics

表 6-132 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32

表 6-133 Query 参数

参数	是否必选	参数类型	描述
timezone	是	String	时区

请求参数

表 6-134 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-135 请求 Body 参数

参数	是否必选	参数类型	描述
start_time	是	Long	开始时间时间戳，毫秒时间，最多支持30天范围内的查询
end_time	是	Long	结束时间时间戳，毫秒时间

参数	是否必选	参数类型	描述
period	是	Integer	查询时间间隔，单位为小时，范围为1-24
resource_type	是	String	资源类型，log_group / log_stream / tenant
search_type	是	String	查询流量类型值为：write, index, storage, basicTransfer, seniorTransfer, coldStorage
resource_id	否	String	资源ID。 当资源类型为log_group时，resource_id是日志组id； 当资源类型为log_stream时，resource_id是日志流id。

响应参数

状态码：200

表 6-136 响应 Body 参数

参数	参数类型	描述
results	Array of Resultits objects	响应结果

表 6-137 Resultits

参数	参数类型	描述
timestamp	Long	时间戳，毫秒时间
value	Double	流量，byte

状态码：400

表 6-138 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36

参数	参数类型	描述
errorMessage	String	错误描述 最小长度：2 最大长度：512

状态码：500

表 6-139 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误码 最小长度：8 最大长度：36
errorMessage	String	错误描述 最小长度：2 最大长度：512

请求示例

按时间段统计查询资源

```
v2/2a473356cca5487f8373be891bffc1cf/lts/timeline-traffic-statistics?timezone=Asia/Shanghai
{
  "start_time": 1668614400000,
  "end_time": 1668787200000,
  "search_type": "write",
  "period": 1,
  "resource_type": "tenant"
}
```

响应示例

状态码：200

查询资源成功

```
{
  "results": [ {
    "timestamp": 1669046400000,
    "value": 8.24859442E7
  }, {
    "timestamp": 1669071600000,
    "value": 0
  }, {
    "timestamp": 1669161600000,
    "value": 9.06895742E7
  }, {
    "timestamp": 1669215600000,
    "value": 8.81524816E7
  }
]
```

状态码：400

BadRequest 非法请求建议根据error_msg直接修改该请求

```
{
  "errorCode": "LTS.0009",
  "errorMessage": "resource_id must not be empty"
}
```

状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
  "errorCode": "LTS.0203",
  "errorMessage": "Internal Server Error"
}
```

SDK 代码示例

SDK代码示例如下。

Java

按时间段统计查询资源

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListTimeLineTrafficStatisticsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListTimeLineTrafficStatisticsRequest request = new ListTimeLineTrafficStatisticsRequest();
        TimelineTrafficStatisticsRequestBody body = new TimelineTrafficStatisticsRequestBody();
        body.withSearchType("write");
        body.withResourceType("tenant");
        body.withPeriod(1);
        body.withEndTime(1668787200000L);
        body.withStartTime(1668614400000L);
        request.withBody(body);
    }
}
```

```
try {
    ListTimeLineTrafficStatisticsResponse response = client.listTimeLineTrafficStatistics(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

按时间段统计查询资源

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTimeLineTrafficStatisticsRequest()
        request.body = TimelineTrafficStatisticsRequestBody(
            search_type="write",
            resource_type="tenant",
            period=1,
            end_time=1668787200000,
            start_time=1668614400000
        )
        response = client.list_time_line_traffic_statistics(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

按时间段统计查询资源

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListTimeLineTrafficStatisticsRequest{}
    request.Body = &model.TimelineTrafficStatisticsRequestBody{
        SearchType: "write",
        ResourceType: "tenant",
        Period: int32(1),
        EndTime: int64(1668787200000),
        StartTime: int64(1668614400000),
    }
    response, err := client.ListTimeLineTrafficStatistics(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	查询资源成功
400	BadRequest 非法请求建议根据error_msg直接修改该请求
500	表明服务端能被请求访问到，但是服务内部出错

错误码

请参见[错误码](#)。

6.4.3 查询日志

功能介绍

该接口用于查询指定日志流下的日志内容。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/
query

表 6-140 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36
log_stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36

请求参数

表 6-141 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-142 请求 Body 参数

参数	是否必选	参数类型	描述
start_time	是	String	搜索起始时间（UTC时间，毫秒级）。 说明 查询时间最大为180天。
end_time	是	String	搜索结束时间（UTC时间，毫秒级）。 说明 查询时间最大为180天。
labels	否	Map<String,String>	日志过滤条件集合，不同日志来源所需字段不同。
keywords	否	String	支持关键词精确搜索。关键词指相邻两个分词符之间的单词，例：error
line_num	否	String	日志单行序列号，第一次查询时不需要此参数，后续分页查询时需要使用，可从上次查询的返回信息中获取。line_num应在start_time 和end_time之间。若已开启自定义时间功能，在使用该字段的同时，还需要增加__time__字段共同进行分页查询。 最小长度：19 最大长度：19

参数	是否必选	参数类型	描述
__time__	否	String	若已开启自定义时间功能，需要使用该字段和line_num字段共同进行分页查询，参数可从上次查询的返回信息中获取。
is_desc	否	Boolean	顺序或者倒序查询，默认为false(顺序查询) 枚举值： ● false
search_type	否	String	首次查询为“init”，分页查询时为“forwards”或者“backwards”，默认为首次查询“init”，与is_desc参数配合进行分页查询。 枚举值： ● forwards
limit	否	Integer	表示每次查询的日志条数，不填时默认为50，建议您设置为100。 最小值： 1 最大值： 5000
highlight	否	Boolean	日志关键词高亮显示，默认为true（高亮显示），false为取消高亮显示。 缺省值： true
is_count	否	Boolean	日志条数统计。默认为false(不统计)，true为统计日志条数。
is_iterative	否	Boolean	日志迭代查询，默认为false（不开启迭代），true为开启迭代。
query	否	String	使用带管道符的sql分析语句进行查询，需要query参数 is_analysis_query为true时生效。

参数	是否必选	参数类型	描述
is_analysis_query	否	Boolean	是否为带管道符的sql分析语句。当该参数为true时，将依照body中的query参数内容进行查询，且body中除start_time与end_time以外的参数失效，分页、排序、查询结果条数等功能请依照sql语法规则实现。查询结果的响应体不同于未启用时的查询方式，将以默认列存的形式返回查询结果。当前仅对内测用户开放。响应示例： {"analysisLogs": [{"field1":"1","field2":"2","field3":"3"}, {"field1":"1","field2":"2","field3":"3"}, {"field1":"1","field2":"2","field3":"3"}]}（管道符为邀测功能，非白名单用户默认为false）

响应参数

状态码：200

表 6-143 响应 Body 参数

参数	参数类型	描述
logs	Array of LogContents objects	日志信息。
count	Integer	日志条数。
isQueryComplete	Boolean	是否查询完成。
analysisLogs	Array of objects	分析日志返回响应体

表 6-144 LogContents

参数	参数类型	描述
content	String	日志原数据。 最小长度：1 最大长度：10000

参数	参数类型	描述
line_num	String	日志单行序列号。 最小长度：19 最大长度：19
labels	Map<String,String>	该条日志包含的 labels,查询到的日志不同所包含的字段不同。

状态码：400

表 6-145 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-146 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-147 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-148 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

- 首次查询日志

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{
  "start_time": 1595659200000,
  "end_time": 1595659500000,
  "labels": {
    "hostName": "ecs-kwxtest"
  },
  "keywords": "log",
  "limit": 10,
  "is_count": true
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 6”、“NO 7”、“NO 8”

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{
  "start_time": 1595659200000,
  "end_time": 1595659500000,
  "labels": {
    "hostName": "ecs-kwxtest"
  },
  "keywords": "log",
```

```
"line_num": "1595659490239433658",  
"is_desc": "false",  
"search_type": "forwards",  
"limit": "3",  
"is_count": true  
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 8”、“NO 7”、“NO 6”

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{  
  "start_time": 1595659200000,  
  "end_time": 1595659500000,  
  "labels": {  
    "hostName": "ecs-kwxtest"  
  },  
  "keywords": "log",  
  "line_num": "1595659490239433658",  
  "is_desc": "true",  
  "search_type": "backwards",  
  "limit": "3",  
  "is_count": true  
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 2”、“NO 3”、“NO 4”

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{  
  "start_time": 1595659200000,  
  "end_time": 1595659500000,  
  "labels": {  
    "hostName": "ecs-kwxtest"  
  },  
  "keywords": "log",  
  "line_num": "1595659490239433658",  
  "is_desc": "false",  
  "search_type": "backwards",  
  "limit": "3",  
  "is_count": true  
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 4”、“NO 3”、“NO 2”

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{  
  "start_time": 1595659200000,  
  "end_time": 1595659500000,  
  "labels": {  
    "hostName": "ecs-kwxtest"  
  },  
  "keywords": "log",  
  "line_num": "1595659490239433658",  
  "is_desc": "true",  
  "search_type": "forwards",  
  "limit": "3",  
  "is_count": true  
}
```

- 分析语句查询示例。

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query
```

```
{  
  "start_time": 1722324332801,  
  "end_time": 1722327932801,  
  "query": "* | select field1,field2 from log",  
}
```

```
"is_analysis_query" : true
}
```

响应示例

状态码：200

请求响应成功。

- 首次查询日志

```
{
  "count" : 32,
  "logs" : [ {
    "content" : "2020-07-25/14:44:42 this <HighLightTag>log</HighLightTag> is Error NO 1",
    "labels" : {
      "hostName" : "ecs-kwxtest",
      "hostIP" : "192.168.0.156",
      "appName" : "default_appname",
      "containerName" : "CONFIG_FILE",
      "clusterName" : "CONFIG_FILE",
      "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName" : "default_procname",
      "clusterId" : "CONFIG_FILE",
      "nameSpace" : "CONFIG_FILE",
      "category" : "LTS"
    },
    "line_num" : "1595659490239433654"
  }, {
    "content" : "2020-07-25/14:44:43 this <HighLightTag>log</HighLightTag> is Error NO 2",
    "labels" : {
      "hostName" : "ecs-kwxtest",
      "hostIP" : "192.168.0.156",
      "appName" : "default_appname",
      "containerName" : "CONFIG_FILE",
      "clusterName" : "CONFIG_FILE",
      "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName" : "default_procname",
      "clusterId" : "CONFIG_FILE",
      "nameSpace" : "CONFIG_FILE",
      "category" : "LTS"
    },
    "line_num" : "1595659490239433655"
  }, {
    "content" : "2020-07-25/14:44:44 this <HighLightTag>log</HighLightTag> is Error NO 3",
    "labels" : {
      "hostName" : "ecs-kwxtest",
      "hostIP" : "192.168.0.156",
      "appName" : "default_appname",
      "containerName" : "CONFIG_FILE",
      "clusterName" : "CONFIG_FILE",
      "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName" : "default_procname",
      "clusterId" : "CONFIG_FILE",
      "nameSpace" : "CONFIG_FILE",
      "category" : "LTS"
    },
    "line_num" : "1595659490239433656"
  }, {
    "content" : "2020-07-25/14:44:45 this <HighLightTag>log</HighLightTag> is Error NO 4",
    "labels" : {
      "hostName" : "ecs-kwxtest",
      "hostIP" : "192.168.0.156",
      "appName" : "default_appname",
      "containerName" : "CONFIG_FILE",
      "clusterName" : "CONFIG_FILE",
      "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName" : "default_procname",
      "clusterId" : "CONFIG_FILE",

```

```
    "nameSpace" : "CONFIG_FILE",
    "category" : "LTS"
  },
  "line_num" : "1595659490239433657"
}, {
  "content" : "2020-07-25/14:44:46 this <HighLightTag>log</HighLightTag> is Error NO 5",
  "labels" : {
    "hostName" : "ecs-kwxtest",
    "hostIP" : "192.168.0.156",
    "appName" : "default_appname",
    "containerName" : "CONFIG_FILE",
    "clusterName" : "CONFIG_FILE",
    "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName" : "default_procname",
    "clusterId" : "CONFIG_FILE",
    "nameSpace" : "CONFIG_FILE",
    "category" : "LTS"
  },
  "line_num" : "1595659490239433658"
}, {
  "content" : "2020-07-25/14:44:47 this <HighLightTag>log</HighLightTag> is Error NO 6",
  "labels" : {
    "hostName" : "ecs-kwxtest",
    "hostIP" : "192.168.0.156",
    "appName" : "default_appname",
    "containerName" : "CONFIG_FILE",
    "clusterName" : "CONFIG_FILE",
    "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName" : "default_procname",
    "clusterId" : "CONFIG_FILE",
    "nameSpace" : "CONFIG_FILE",
    "category" : "LTS"
  },
  "line_num" : "1595659490239433659"
}, {
  "content" : "2020-07-25/14:44:48 this <HighLightTag>log</HighLightTag> is Error NO 7",
  "labels" : {
    "hostName" : "ecs-kwxtest",
    "hostIP" : "192.168.0.156",
    "appName" : "default_appname",
    "containerName" : "CONFIG_FILE",
    "clusterName" : "CONFIG_FILE",
    "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName" : "default_procname",
    "clusterId" : "CONFIG_FILE",
    "nameSpace" : "CONFIG_FILE",
    "category" : "LTS"
  },
  "line_num" : "1595659490239433660"
}, {
  "content" : "2020-07-25/14:44:49 this <HighLightTag>log</HighLightTag> is Error NO 8",
  "labels" : {
    "hostName" : "ecs-kwxtest",
    "hostIP" : "192.168.0.156",
    "appName" : "default_appname",
    "containerName" : "CONFIG_FILE",
    "clusterName" : "CONFIG_FILE",
    "hostId" : "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName" : "default_procname",
    "clusterId" : "CONFIG_FILE",
    "nameSpace" : "CONFIG_FILE",
    "category" : "LTS"
  },
  "line_num" : "1595659490239433661"
}, {
  "content" : "2020-07-25/14:44:50 this <HighLightTag>log</HighLightTag> is Error NO 9",
  "labels" : {
    "hostName" : "ecs-kwxtest",
    "hostIP" : "192.168.0.156",
```

```
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659490839420574"
}, {
"content": "2020-07-25/14:44:51 this <HighLightTag>log</HighLightTag> is Error NO 10",
"labels": {
"hostName": "ecs-kwxtest",
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659491839412667"
}
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 6”、“NO 7”、“NO 8”

```
{
"count": 32,
"logs": [ {
"content": "2020-07-25/14:44:47 this <HighLightTag>log</HighLightTag> is Error NO 6",
"labels": {
"hostName": "ecs-kwxtest",
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659490239433659"
}, {
"content": "2020-07-25/14:44:48 this <HighLightTag>log</HighLightTag> is Error NO 7",
"labels": {
"hostName": "ecs-kwxtest",
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659490239433660"
}, {
"content": "2020-07-25/14:44:49 this <HighLightTag>log</HighLightTag> is Error NO 8",
"labels": {
"hostName": "ecs-kwxtest",
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
```

```

    "clusterName": "CONFIG_FILE",
    "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName": "default_procname",
    "clusterId": "CONFIG_FILE",
    "nameSpace": "CONFIG_FILE",
    "category": "LTS"
  },
  "line_num": "1595659490239433661"
}]
}

```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 8”、“NO 7”、“NO 6”

```

{
  "count": 32,
  "logs": [ {
    "content": "2020-07-25/14:44:49 this <HighLightTag>log</HighLightTag> is Error NO 8",
    "labels": {
      "hostName": "ecs-kwxtest",
      "hostIP": "192.168.0.156",
      "appName": "default_appname",
      "containerName": "CONFIG_FILE",
      "clusterName": "CONFIG_FILE",
      "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName": "default_procname",
      "clusterId": "CONFIG_FILE",
      "nameSpace": "CONFIG_FILE",
      "category": "LTS"
    },
    "line_num": "1595659490239433661"
  }, {
    "content": "2020-07-25/14:44:48 this <HighLightTag>log</HighLightTag> is Error NO 7",
    "labels": {
      "hostName": "ecs-kwxtest",
      "hostIP": "192.168.0.156",
      "appName": "default_appname",
      "containerName": "CONFIG_FILE",
      "clusterName": "CONFIG_FILE",
      "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName": "default_procname",
      "clusterId": "CONFIG_FILE",
      "nameSpace": "CONFIG_FILE",
      "category": "LTS"
    },
    "line_num": "1595659490239433660"
  }, {
    "content": "2020-07-25/14:44:47 this <HighLightTag>log</HighLightTag> is Error NO 6",
    "labels": {
      "hostName": "ecs-kwxtest",
      "hostIP": "192.168.0.156",
      "appName": "default_appname",
      "containerName": "CONFIG_FILE",
      "clusterName": "CONFIG_FILE",
      "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName": "default_procname",
      "clusterId": "CONFIG_FILE",
      "nameSpace": "CONFIG_FILE",
      "category": "LTS"
    },
    "line_num": "1595659490239433659"
  }
  ]
}

```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 2”、“NO 3”、“NO 4”

```

{
  "count": 32,
  "logs": [ {
    "content": "2020-07-25/14:44:43 this <HighLightTag>log</HighLightTag> is Error NO 2",

```

```
"labels": {
  "hostName": "ecs-kwxtest",
  "hostIP": "192.168.0.156",
  "appName": "default_appname",
  "containerName": "CONFIG_FILE",
  "clusterName": "CONFIG_FILE",
  "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
  "podName": "default_procname",
  "clusterId": "CONFIG_FILE",
  "nameSpace": "CONFIG_FILE",
  "category": "LTS"
},
"line_num": "1595659490239433655"
}, {
  "content": "2020-07-25/14:44:44 this <HighLightTag>log</HighLightTag> is Error NO 3",
  "labels": {
    "hostName": "ecs-kwxtest",
    "hostIP": "192.168.0.156",
    "appName": "default_appname",
    "containerName": "CONFIG_FILE",
    "clusterName": "CONFIG_FILE",
    "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName": "default_procname",
    "clusterId": "CONFIG_FILE",
    "nameSpace": "CONFIG_FILE",
    "category": "LTS"
  },
  "line_num": "1595659490239433656"
}, {
  "content": "2020-07-25/14:44:45 this <HighLightTag>log</HighLightTag> is Error NO 4",
  "labels": {
    "hostName": "ecs-kwxtest",
    "hostIP": "192.168.0.156",
    "appName": "default_appname",
    "containerName": "CONFIG_FILE",
    "clusterName": "CONFIG_FILE",
    "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
    "podName": "default_procname",
    "clusterId": "CONFIG_FILE",
    "nameSpace": "CONFIG_FILE",
    "category": "LTS"
  },
  "line_num": "1595659490239433657"
}
}]
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 4”、“NO 3”、“NO 2”

```
{
  "count": 32,
  "logs": [ {
    "content": "2020-07-25/14:44:45 this <HighLightTag>log</HighLightTag> is Error NO 4",
    "labels": {
      "hostName": "ecs-kwxtest",
      "hostIP": "192.168.0.156",
      "appName": "default_appname",
      "containerName": "CONFIG_FILE",
      "clusterName": "CONFIG_FILE",
      "hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
      "podName": "default_procname",
      "clusterId": "CONFIG_FILE",
      "nameSpace": "CONFIG_FILE",
      "category": "LTS"
    },
    "line_num": "1595659490239433657"
  }, {
    "content": "2020-07-25/14:44:44 this <HighLightTag>log</HighLightTag> is Error NO 3",
    "labels": {
      "hostName": "ecs-kwxtest",
```

```
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659490239433656"
}, {
"content": "2020-07-25/14:44:43 this <HighLightTag>log</HighLightTag> is Error NO 2",
"labels": {
"hostName": "ecs-kwxtest",
"hostIP": "192.168.0.156",
"appName": "default_appname",
"containerName": "CONFIG_FILE",
"clusterName": "CONFIG_FILE",
"hostId": "9787ef31-fd7b-4eff-ba71-72d580f11f55",
"podName": "default_procname",
"clusterId": "CONFIG_FILE",
"nameSpace": "CONFIG_FILE",
"category": "LTS"
},
"line_num": "1595659490239433655"
}
}]
}
```

状态码：400

BadRequest。非法请求或查询语句错误。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
"error_code": "LTS.0009",
"error_msg": "Failed to validate the request body"
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
"error_code": "LTS.0003",
"error_msg": "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
"error_code": "LTS.0001",
"error_msg": "Invalid projectId"
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
"error_code": "LTS.0202",
"error_msg": "Failed to query lts log"
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 首次查询日志

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        Map<String, String> listbodyLabels = new HashMap<>();
        listbodyLabels.put("hostName", "ecs-kwxtest");
        body.withLimit(10);
        body.withKeywords("log");
        body.withIsCount(true);
        body.withLabels(listbodyLabels);
        body.withEndTime("1595659500000");
        body.withStartTime("1595659200000");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
        }
    }
}
```

```
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 6”、“NO 7”、“NO 8”

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        Map<String, String> listbodyLabels = new HashMap<>();
        listbodyLabels.put("hostName", "ecs-kwxtest");
        body.withLimit(3);
        body.withSearchType(QueryLtsLogParams.SearchTypeEnum.fromValue("forwards"));
        body.withIsDesc(false);
        body.withLineNum("1595659490239433658");
        body.withKeywords("log");
        body.withIsCount(true);
        body.withLabels(listbodyLabels);
        body.withEndTime("1595659500000");
        body.withStartTime("1595659200000");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
        }
    }
}
```

```
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 8”、“NO 7”、“NO 6”

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        Map<String, String> listbodyLabels = new HashMap<>();
        listbodyLabels.put("hostName", "ecs-kwxtest");
        body.withLimit(3);
        body.withSearchType(QueryLtsLogParams.SearchTypeEnum.fromValue("backwards"));
        body.withIsDesc(true);
        body.withLineNum("1595659490239433658");
        body.withKeywords("log");
        body.withIsCount(true);
        body.withLabels(listbodyLabels);
        body.withEndTime("1595659500000");
        body.withStartTime("1595659200000");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

```
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 2”、“NO 3”、“NO 4”

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        Map<String, String> listbodyLabels = new HashMap<>();
        listbodyLabels.put("hostName", "ecs-kwxtest");
        body.withLimit(3);
        body.withSearchType(QueryLtsLogParams.SearchTypeEnum.fromValue("backwards"));
        body.withIsDesc(false);
        body.withLineNum("1595659490239433658");
        body.withKeywords("log");
        body.withIsCount(true);
        body.withLabels(listbodyLabels);
        body.withEndTime("1595659500000");
        body.withStartTime("1595659200000");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
```

```
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 分页查询, 以包含 “NO 5” 的日志为起始点, 查询 “NO 4”、“NO 3”、“NO 2”

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.Map;
import java.util.HashMap;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        Map<String, String> listbodyLabels = new HashMap<>();
        listbodyLabels.put("hostName", "ecs-kwxtest");
        body.withLimit(3);
        body.withSearchType(QueryLtsLogParams.SearchTypeEnum.fromValue("forwards"));
        body.withIsDesc(true);
        body.withLineNum("1595659490239433658");
        body.withKeywords("log");
        body.withIsCount(true);
        body.withLabels(listbodyLabels);
        body.withEndTime("1595659500000");
        body.withStartTime("1595659200000");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 分析语句查询示例。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListLogsRequest request = new ListLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsLogParams body = new QueryLtsLogParams();
        body.withIsAnalysisQuery(true);
        body.withQuery("* | select field1,field2 from log");
        body.withEndTime("1722327932801");
        body.withStartTime("1722324332801");
        request.withBody(body);
        try {
            ListLogsResponse response = client.listLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}  
}
```

Python

- 首次查询日志

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsklts.v2.region.lts_region import LtsRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsklts.v2 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    # environment variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before  
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    # environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = LtsClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ListLogsRequest()  
        request.log_group_id = "{log_group_id}"  
        request.log_stream_id = "{log_stream_id}"  
        listLabelsbody = {  
            "hostName": "ecs-kwxtest"  
        }  
        request.body = QueryLtsLogParams(  
            limit=10,  
            keywords="log",  
            is_count=True,  
            labels=listLabelsbody,  
            end_time="1595659500000",  
            start_time="1595659200000"  
        )  
        response = client.list_logs(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 6”、“NO 7”、“NO 8”

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsklts.v2.region.lts_region import LtsRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsklts.v2 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(LtsRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListLogsRequest()
    request.log_group_id = "{log_group_id}"
    request.log_stream_id = "{log_stream_id}"
    listLabelsbody = {
        "hostName": "ecs-kwxtest"
    }
    request.body = QueryLtsLogParams(
        limit=3,
        search_type="forwards",
        is_desc=False,
        line_num="1595659490239433658",
        keywords="log",
        is_count=True,
        labels=listLabelsbody,
        end_time="1595659500000",
        start_time="1595659200000"
    )
    response = client.list_logs(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 8”、“NO 7”、“NO 6”

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
try:
    request = ListLogsRequest()
    request.log_group_id = "{log_group_id}"
    request.log_stream_id = "{log_stream_id}"
    listLabelsbody = {
        "hostName": "ecs-kwxtest"
    }
    request.body = QueryLtsLogParams(
        limit=3,
        search_type="backwards",
        is_desc=True,
        line_num="1595659490239433658",
        keywords="log",
        is_count=True,
        labels=listLabelsbody,
        end_time="1595659500000",
        start_time="1595659200000"
    )
    response = client.list_logs(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 2”、“NO 3”、“NO 4”

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListLogsRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        listLabelsbody = {
            "hostName": "ecs-kwxtest"
        }
        request.body = QueryLtsLogParams(
            limit=3,
            search_type="backwards",
            is_desc=False,
            line_num="1595659490239433658",
            keywords="log",
            is_count=True,
```

```
        labels=listLabelsbody,
        end_time="1595659500000",
        start_time="1595659200000"
    )
    response = client.list_logs(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 4”、“NO 3”、“NO 2”

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListLogsRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        listLabelsbody = {
            "hostName": "ecs-kwxtest"
        }
        request.body = QueryLtsLogParams(
            limit=3,
            search_type="forwards",
            is_desc=True,
            line_num="1595659490239433658",
            keywords="log",
            is_count=True,
            labels=listLabelsbody,
            end_time="1595659500000",
            start_time="1595659200000"
        )
        response = client.list_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 分析语句查询示例。

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListLogsRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        request.body = QueryLtsLogParams(
            is_analysis_query=True,
            query="*" | select field1,field2 from log",
            end_time="1722327932801",
            start_time="1722324332801"
        )
        response = client.list_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 首次查询日志

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

```
WithProjectId(projectId).
Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListLogsRequest{}
request.LogGroupId = "{log_group_id}"
request.LogStreamId = "{log_stream_id}"
var listLabelsbody = map[string]string{
    "hostName": "ecs-kwxtest",
}
}
limitQueryLtsLogParams:= int32(10)
keywordsQueryLtsLogParams:= "log"
isCountQueryLtsLogParams:= true
request.Body = &model.QueryLtsLogParams{
    Limit: &limitQueryLtsLogParams,
    Keywords: &keywordsQueryLtsLogParams,
    IsCount: &isCountQueryLtsLogParams,
    Labels: listLabelsbody,
    EndTime: "1595659500000",
    StartTime: "1595659200000",
}
}
response, err := client.ListLogs(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 6”、“NO 7”、“NO 8”

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.ListLogsRequest{}
request.LogGroupId = "{log_group_id}"
request.LogStreamId = "{log_stream_id}"
var listLabelsbody = map[string]string{
    "hostName": "ecs-kwxtest",
}
limitQueryLtsLogParams:= int32(3)
searchTypeQueryLtsLogParams:= model.GetQueryLtsLogParamsSearchTypeEnum().FORWARDS
isDescQueryLtsLogParams:= false
lineNumQueryLtsLogParams:= "1595659490239433658"
keywordsQueryLtsLogParams:= "log"
isCountQueryLtsLogParams:= true
request.Body = &model.QueryLtsLogParams{
    Limit: &limitQueryLtsLogParams,
    SearchType: &searchTypeQueryLtsLogParams,
    IsDesc: &isDescQueryLtsLogParams,
    LineNum: &lineNumQueryLtsLogParams,
    Keywords: &keywordsQueryLtsLogParams,
    IsCount: &isCountQueryLtsLogParams,
    Labels: listLabelsbody,
    EndTime: "1595659500000",
    StartTime: "1595659200000",
}
response, err := client.ListLogs(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 8”、“NO 7”、“NO 6”

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListLogsRequest{}
    request.LogGroupId = "{log_group_id}"
    request.LogStreamId = "{log_stream_id}"
```

```
var listLabelsbody = map[string]string{
    "hostName": "ecs-kwxtest",
}
limitQueryLtsLogParams:= int32(3)
searchTypeQueryLtsLogParams:= model.GetQueryLtsLogParamsSearchTypeEnum().BACKWARDS
isDescQueryLtsLogParams:= true
lineNumQueryLtsLogParams:= "1595659490239433658"
keywordsQueryLtsLogParams:= "log"
isCountQueryLtsLogParams:= true
request.Body = &model.QueryLtsLogParams{
    Limit: &limitQueryLtsLogParams,
    SearchType: &searchTypeQueryLtsLogParams,
    IsDesc: &isDescQueryLtsLogParams,
    LineNum: &lineNumQueryLtsLogParams,
    Keywords: &keywordsQueryLtsLogParams,
    IsCount: &isCountQueryLtsLogParams,
    Labels: listLabelsbody,
    EndTime: "1595659500000",
    StartTime: "1595659200000",
}
response, err := client.ListLogs(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 2”、“NO 3”、“NO 4”

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListLogsRequest{}
    request.LogGroupId = "{log_group_id}"
    request.LogStreamId = "{log_stream_id}"
    var listLabelsbody = map[string]string{
        "hostName": "ecs-kwxtest",
    }
}
```

```
limitQueryLtsLogParams:= int32(3)
searchTypeQueryLtsLogParams:= model.GetQueryLtsLogParamsSearchTypeEnum().BACKWARDS
isDescQueryLtsLogParams:= false
lineNumQueryLtsLogParams:= "1595659490239433658"
keywordsQueryLtsLogParams:= "log"
isCountQueryLtsLogParams:= true
request.Body = &model.QueryLtsLogParams{
    Limit: &limitQueryLtsLogParams,
    SearchType: &searchTypeQueryLtsLogParams,
    IsDesc: &isDescQueryLtsLogParams,
    LineNum: &lineNumQueryLtsLogParams,
    Keywords: &keywordsQueryLtsLogParams,
    IsCount: &isCountQueryLtsLogParams,
    Labels: listLabelsbody,
    EndTime: "1595659500000",
    StartTime: "1595659200000",
}
response, err := client.ListLogs(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 分页查询, 以包含“NO 5”的日志为起始点, 查询“NO 4”、“NO 3”、“NO 2”

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListLogsRequest{}
    request.LogGroupId = "{log_group_id}"
    request.LogStreamId = "{log_stream_id}"
    var listLabelsbody = map[string]string{
        "hostName": "ecs-kwxtest",
    }
}
limitQueryLtsLogParams:= int32(3)
searchTypeQueryLtsLogParams:= model.GetQueryLtsLogParamsSearchTypeEnum().FORWARDS
isDescQueryLtsLogParams:= true
```

```
lineNumQueryLtsLogParams:= "1595659490239433658"  
keywordsQueryLtsLogParams:= "log"  
isCountQueryLtsLogParams:= true  
request.Body = &model.QueryLtsLogParams{  
    Limit: &limitQueryLtsLogParams,  
    SearchType: &searchTypeQueryLtsLogParams,  
    IsDesc: &isDescQueryLtsLogParams,  
    LineNum: &lineNumQueryLtsLogParams,  
    Keywords: &keywordsQueryLtsLogParams,  
    IsCount: &isCountQueryLtsLogParams,  
    Labels: listLabelsbody,  
    EndTime: "1595659500000",  
    StartTime: "1595659200000",  
}  
response, err := client.ListLogs(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

- 分析语句查询示例。

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    // environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before  
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    // environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := lts.NewLtsClient(  
        lts.LtsClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListLogsRequest{  
        request.LogGroupId = "{log_group_id}"  
        request.LogStreamId = "{log_stream_id}"  
        isAnalysisQueryQueryLtsLogParams:= true  
        queryQueryLtsLogParams:= "*" | select field1,field2 from log"  
        request.Body = &model.QueryLtsLogParams{  
            IsAnalysisQuery: &isAnalysisQueryQueryLtsLogParams,  
            Query: &queryQueryLtsLogParams,  
            EndTime: "1722327932801",  
            StartTime: "1722324332801",  
        }  
    }  
    response, err := client.ListLogs(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    }  
}
```

```
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求响应成功。
400	BadRequest。非法请求或查询语句错误。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败，请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.4.4 查询结构化日志（即将下线不推荐使用）

功能介绍

该接口用于查询指定日志流下的结构化日志内容。（该接口将于2024年7月30日下线）

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/struct-content/query

表 6-149 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：32 最大长度：32
log_group_id	是	String	日志组ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36
log_stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID ， 获取账号ID ， 日志组ID 、 日志流ID 最小长度：36 最大长度：36

请求参数

表 6-150 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-151 请求 Body 参数

参数	是否必选	参数类型	描述
start_time	是	String	搜索起始时间（UTC时间，毫秒级）。

参数	是否必选	参数类型	描述
end_time	是	String	搜索结束时间（UTC时间，毫秒级）。
sql_expression	否	String	支持SQL语句搜索，目前支持"GROUP BY", "LIKE"和"WHERE"。默认为"SELECT *" 最小长度：0 最大长度：100
original_content	否	Boolean	返回内容中是否包含原始日志，默认为false。 枚举值： • false

响应参数

状态码：200

表 6-152 响应 Body 参数

参数	参数类型	描述
struct_logs	Array of StructLogContents objects	日志信息。

表 6-153 StructLogContents

参数	参数类型	描述
log_content	String	日志原数据。 最小长度：1 最大长度：10000
line_num	String	日志单行序列号。 最小长度：19 最大长度：19

状态码：400

表 6-154 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：401

表 6-155 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：403

表 6-156 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-157 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

请求示例

- 查询结构化日志

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/struct-content/query
```

- 当 “sql_expression” 参数填写为select *

```
v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/struct-content/query
```

```
{
  "start_time": "1595811590539",
  "end_time": "1595811593539",
  "original_content": "true",
  "sql_expression": "select *"
}
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4)

```
{
  "start_time": "1595811590539",
  "end_time": "1595811593539",
  "sql_expression": "select count(test3), sum(test4)"
}
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4) group by(test1)

```
{
  "start_time": "1595811590539",
  "end_time": "1595811593539",
  "sql_expression": "select count(test3), sum(test4) group by(test1)"
}
```

响应示例

状态码：200

请求响应成功。

- 当 “sql_expression” 参数填写为select *

```
{
  "struct_logs": [ {
    "test4": "151687",
    "test2": "08:59:53",
    "test3": "this",
    "line_num": "1595811593539371695",
    "log_content": "2020-07-27/08:59:53 this log is Error NO 151687",
    "test1": "2020-07-27"
  }, {
```

```
"test4" : "151686",
"test2" : "08:59:52",
"test3" : "this",
"line_num" : "1595811592539361171",
"log_content" : "2020-07-27/08:59:52 this log is Error NO 151686",
"test1" : "2020-07-27"
}, {
"test4" : "151685",
"test2" : "08:59:51",
"test3" : "this",
"line_num" : "1595811591539422860",
"log_content" : "2020-07-27/08:59:51 this log is Error NO 151685",
"test1" : "2020-07-27"
}, {
"test4" : "151684",
"test2" : "08:59:50",
"test3" : "this",
"line_num" : "1595811590539454127",
"log_content" : "2020-07-27/08:59:50 this log is Error NO 151684",
"test1" : "2020-07-27"
}
}]
}
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4)

```
{
"struct_logs" : [ {
"cnt(test3)" : "4",
"sum(test4)" : "606742.000"
}
]
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4) group by(test1)

```
{
"struct_logs" : [ {
"cnt(test3)" : "4",
"sum(test4)" : "606742.000",
"test1" : "2020-07-27"
}
]
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4) group by(test1)

```
{
"struct_logs" : [ {
"cnt(test3)" : "4",
"sum(test4)" : "606742.000",
"test1" : "2020-07-27"
}
]
```

状态码：400

BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。

```
{
"error_code" : "LTS.0009",
"error_msg" : "Failed to validate the request body"
}
```

状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
"error_code" : "LTS.0414",
"error_msg" : "Invalid token"
}
```

状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
  "error_code" : "LTS.0001",
  "error_msg" : "Invalid projectId"
}
```

状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
  "error_code" : "LTS.0202",
  "error_msg" : "Failed to query lts struct log"
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 当“sql_expression”参数填写为select *

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListQueryStructuredLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListQueryStructuredLogsRequest request = new ListQueryStructuredLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
```

```
QueryLtsStructLogParams body = new QueryLtsStructLogParams();
body.withOriginalContent(true);
body.withSqlExpression("select *");
body.withEndTime("1595811593539");
body.withStartTime("1595811590539");
request.withBody(body);
try {
    ListQueryStructuredLogsResponse response = client.listQueryStructuredLogs(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- 当 “sql_expression” 参数填写为 select count(test3), sum(test4)

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListQueryStructuredLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListQueryStructuredLogsRequest request = new ListQueryStructuredLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsStructLogParams body = new QueryLtsStructLogParams();
        body.withSqlExpression("select count(test3), sum(test4)");
        body.withEndTime("1595811593539");
        body.withStartTime("1595811590539");
        request.withBody(body);
        try {
            ListQueryStructuredLogsResponse response = client.listQueryStructuredLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 当“sql_expression”参数填写为select count(test3), sum(test4) group by(test1)

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListQueryStructuredLogsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        LtsClient client = LtsClient.newBuilder()
            .withCredential(auth)
            .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
            .build();
        ListQueryStructuredLogsRequest request = new ListQueryStructuredLogsRequest();
        request.withLogGroupId("{log_group_id}");
        request.withLogStreamId("{log_stream_id}");
        QueryLtsStructLogParams body = new QueryLtsStructLogParams();
        body.withSqlExpression("select count(test3), sum(test4) group by(test1)");
        body.withEndTime("1595811593539");
        body.withStartTime("1595811590539");
        request.withBody(body);
        try {
            ListQueryStructuredLogsResponse response = client.listQueryStructuredLogs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
        }
    }
}
```

```
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

- 当 “sql_expression” 参数填写为select *

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListQueryStructuredLogsRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        request.body = QueryLtsStructLogParams(
            original_content=True,
            sql_expression="select *",
            end_time="1595811593539",
            start_time="1595811590539"
        )
        response = client.list_query_structured_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4)

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
```

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(LtsRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListQueryStructuredLogsRequest()
    request.log_group_id = "{log_group_id}"
    request.log_stream_id = "{log_stream_id}"
    request.body = QueryLtsStructLogParams(
        sql_expression="select count(test3), sum(test4)",
        end_time="1595811593539",
        start_time="1595811590539"
    )
    response = client.list_query_structured_logs(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 当“sql_expression”参数填写为select count(test3), sum(test4) group by(test1)

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = LtsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(LtsRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListQueryStructuredLogsRequest()
        request.log_group_id = "{log_group_id}"
        request.log_stream_id = "{log_stream_id}"
        request.body = QueryLtsStructLogParams(
            sql_expression="select count(test3), sum(test4) group by(test1)",
            end_time="1595811593539",
            start_time="1595811590539"
        )
        response = client.list_query_structured_logs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
```

```
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

Go

- 当 “sql_expression” 参数填写为select *

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := lts.NewLtsClient(
        lts.LtsClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListQueryStructuredLogsRequest{}
    request.LogGroupId = "{log_group_id}"
    request.LogStreamId = "{log_stream_id}"
    request.OriginalContentQueryLtsStructLogParams = true
    request.SqlExpressionQueryLtsStructLogParams = "select *"
    request.Body = &model.QueryLtsStructLogParams{
        OriginalContent: &originalContentQueryLtsStructLogParams,
        SqlExpression: &sqlExpressionQueryLtsStructLogParams,
        EndTime: "1595811593539",
        StartTime: "1595811590539",
    }
    response, err := client.ListQueryStructuredLogs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 当 “sql_expression” 参数填写为select count(test3), sum(test4)

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)
```

```
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before  
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := lts.NewLtsClient(  
        lts.LtsClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build()  
    )  
  
    request := &model.ListQueryStructuredLogsRequest{}  
    request.LogGroupId = "{log_group_id}"  
    request.LogStreamId = "{log_stream_id}"  
    sqlExpressionQueryLtsStructLogParams := "select count(test3), sum(test4)"  
    request.Body = &model.QueryLtsStructLogParams{  
        SqlExpression: &sqlExpressionQueryLtsStructLogParams,  
        EndTime: "1595811593539",  
        StartTime: "1595811590539",  
    }  
    response, err := client.ListQueryStructuredLogs(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

- 当“sql_expression”参数填写为select count(test3), sum(test4) group by(test1)

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before  
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).
```

```
Build()

client := lts.NewLtsClient(
    lts.LtsClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListQueryStructuredLogsRequest{}
request.LogGroupId = "{log_group_id}"
request.LogStreamId = "{log_stream_id}"
sqlExpressionQueryLtsStructLogParams:= "select count(test3), sum(test4) group by(test1)"
request.Body = &model.QueryLtsStructLogParams{
    SqlExpression: &sqlExpressionQueryLtsStructLogParams,
    EndTime: "1595811593539",
    StartTime: "1595811590539",
}
response, err := client.ListQueryStructuredLogs(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	请求响应成功。
400	BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
401	AuthFailed。鉴权失败，请确认token后再次请求。
403	Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
500	InternalServerError。 表明服务端能被请求访问到，但是服务内部出错。

错误码

请参见[错误码](#)。

6.4.5 查询结构化日志（新版）

功能介绍

该接口用于查询指定日志流下的结构化日志内容（新版）。

说明

新版查询功能支持的局点信息请参考用户指南-日志可视化里的可视化章节。数据聚合结果不超过100亿条。仅支持分析最近30天内的数据，30天以上的数据不支持SQL分析。如您有更大的使用需求，请提工单申请。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/streams/{log_stream_id}/struct-content/query

表 6-158 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：32 最大长度：32
log_stream_id	是	String	日志流ID，获取方式请参见： 获取项目ID，获取账号ID，日志组ID、日志流ID 最小长度：36 最大长度：36

请求参数

表 6-159 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM服务获取的用户Token，获取方式请参见： 获取用户Token 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=UTF-8。 最小长度：30 最大长度：30

表 6-160 请求 Body 参数

参数	是否必选	参数类型	描述
query	是	String	sql语句字符串。 最小长度： 0 最大长度： 102400
format	是	String	查询结果格式。当前仅支持："k-v"。
time_range	是	TimeRange object	时间范围信息。
whether_to_rows	否	Boolean	返回数据格式，是否为行数据，默认为false。

表 6-161 TimeRange

参数	是否必选	参数类型	描述
sql_time_zone	否	String	时区信息，默认为“UTC”。
start_time	是	Long	搜索起始时间（UTC时间，毫秒级）。
end_time	是	Long	搜索结束时间（UTC时间，毫秒级）。
start_time_gt	否	Boolean	搜索是否包含起始时间点，默认为false。
end_time_lt	否	Boolean	搜索是否包含结束时间点，默认为false。

响应参数

状态码：400

表 6-162 响应 Body 参数

参数	参数类型	描述
message	message object	接口调用信息。

状态码：401

表 6-163 响应 Body 参数

参数	参数类型	描述
message	Message401 object	接口调用信息。

表 6-164 Message401

参数	参数类型	描述
code	String	错误码。 枚举值： • LTS.0001
details	String	调用失败响应信息描述。 枚举值： • project verify error

状态码：403

表 6-165 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403
error_msg	String	调用失败响应信息描述。 枚举值： • Invalid projectId

状态码：500

表 6-166 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。
error_msg	String	调用失败响应信息描述。

请求示例

- 查询结构化日志

POST https://{endpoint}/v2/{project_id}/streams/{log_stream_id}/struct-content/query

- 当 “query” 参数填写为 select *

```
{
  "query": "select *",
  "format": "k-v",
  "time_range": {
    "sql_time_zone": "UTC",
    "start_time": 1613806109000,
    "end_time": 1613806110000,
    "start_time_gt": false,
    "end_time_lt": false
  }
}
```

- 当 “query” 参数填写为 select count(test3) as mycount, sum(test4) as mysum

```
{
  "query": "select count(test3) as mycount, sum(test4) as mysum",
  "format": "k-v",
  "time_range": {
    "sql_time_zone": "UTC",
    "start_time": 1613806109000,
    "end_time": 1613806110000,
    "start_time_gt": false,
    "end_time_lt": false
  }
}
```

- 当 “query” 参数填写为 select count(test3), sum(test4) group by(test1)

```
{
  "query": "select count(test3), sum(test4) group by(test1)",
  "format": "k-v",
  "time_range": {
    "sql_time_zone": "UTC",
    "start_time": 1613806109000,
    "end_time": 1613806110000,
    "start_time_gt": false,
    "end_time_lt": false
  }
}
```

- 当 “whether_to_rows” 参数填写为 true

```
{
  "query": "select count(test3), sum(test4) group by(test1)",
  "format": "k-v",
  "time_range": {
    "sql_time_zone": "UTC",
    "start_time": 1613806109000,
    "end_time": 1613806110000,
    "start_time_gt": false,
    "end_time_lt": false
  },
  "whether_to_rows": true
}
```

响应示例

状态码：200

请求响应成功。

- 当 “query” 参数填写为 select *

- ```
{
 "result": ["system context", "global context", "global context", "system context", "system context"]
}
```
- 当 “query” 参数填写为 `select count(test3) as mycount, sum(test4) as mysum`

```
{
 "mycount": [50],
 "mysum": [7006779039596516000]
}
```
  - 当 “query” 参数填写为 `select count(test3) as mycount, sum(test4) as mysum group by(test1)`

```
{
 "mycount": [32, 22],
 "mysum": [-3632218674319473000, -1344225381314777000]
}
```
  - 当 “whether\_to\_rows” 参数填写为 `true`

```
{
 "result": [{
 "mycount": "32",
 "mysum": "-3632218674319473398"
 }, {
 "mycount": "22",
 "mysum": "-1344225381314777019"
 }]
}
```

**状态码：400**

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "message": {
 "code": "LTS.0601",
 "details": "params validator error"
 }
}
```

**状态码：401**

AuthFailed。鉴权失败，请确认token后再次请求。

```
{
 "message": {
 "code": "LTS.0001",
 "details": "project verify error"
 }
}
```

**状态码：403**

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0010",
}
```

```
"error_msg" : "Internal Server Error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 当“query”参数填写为 select \*

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListStructuredLogsWithTimeRangeSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListStructuredLogsWithTimeRangeRequest request = new
ListStructuredLogsWithTimeRangeRequest();
request.withLogStreamId("{log_stream_id}");
QueryLtsStructLogParamsNew body = new QueryLtsStructLogParamsNew();
TimeRange timeRangebody = new TimeRange();
timeRangebody.withSqlTimeZone("UTC")
 .withStartTime(1613806109000L)
 .withEndTime(1613806110000L)
 .withStartTimeGt(false)
 .withEndTimeLt(false);
body.withTimeRange(timeRangebody);
body.withFormat("k-v");
body.withQuery("select *");
request.withBody(body);
try {
 ListStructuredLogsWithTimeRangeResponse response =
client.listStructuredLogsWithTimeRange(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
```

```
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

- 当“query”参数填写为 select count(test3) as mycount, sum(test4) as mysum

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListStructuredLogsWithTimeRangeSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 ListStructuredLogsWithTimeRangeRequest request = new
 ListStructuredLogsWithTimeRangeRequest();
 request.withLogStreamId("{log_stream_id}");
 QueryLtsStructLogParamsNew body = new QueryLtsStructLogParamsNew();
 TimeRange timeRangebody = new TimeRange();
 timeRangebody.withSqlTimeZone("UTC")
 .withStartTime(1613806109000L)
 .withEndTime(1613806110000L)
 .withStartTimeGt(false)
 .withEndTimeLt(false);
 body.withTimeRange(timeRangebody);
 body.withFormat("k-v");
 body.withQuery("select count(test3) as mycount, sum(test4) as mysum");
 request.withBody(body);
 try {
 ListStructuredLogsWithTimeRangeResponse response =
 client.listStructuredLogsWithTimeRange(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 }
 }
}
```

```
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

- 当 “query” 参数填写为 select count(test3), sum(test4) group by(test1)

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListStructuredLogsWithTimeRangeSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 ListStructuredLogsWithTimeRangeRequest request = new
 ListStructuredLogsWithTimeRangeRequest();
 request.withLogStreamId("{log_stream_id}");
 QueryLtsStructLogParamsNew body = new QueryLtsStructLogParamsNew();
 TimeRange timeRangebody = new TimeRange();
 timeRangebody.withSqlTimeZone("UTC")
 .withStartTime(1613806109000L)
 .withEndTime(1613806110000L)
 .withStartTimeGt(false)
 .withEndTimeLt(false);
 body.withTimeRange(timeRangebody);
 body.withFormat("k-v");
 body.withQuery("select count(test3), sum(test4) group by(test1)");
 request.withBody(body);
 try {
 ListStructuredLogsWithTimeRangeResponse response =
 client.listStructuredLogsWithTimeRange(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

- 当 “whether\_to\_rows” 参数填写为 true

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListStructuredLogsWithTimeRangeSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListStructuredLogsWithTimeRangeRequest request = new
ListStructuredLogsWithTimeRangeRequest();
 request.withLogStreamId("{log_stream_id}");
 QueryLtsStructLogParamsNew body = new QueryLtsStructLogParamsNew();
 TimeRange timeRangebody = new TimeRange();
 timeRangebody.withSqlTimeZone("UTC")
 .withStartTime(1613806109000L)
 .withEndTime(1613806110000L)
 .withStartTimeGt(false)
 .withEndTimeLt(false);
 body.withWhetherToRows(true);
 body.withTimeRange(timeRangebody);
 body.withFormat("k-v");
 body.withQuery("select count(test3), sum(test4) group by(test1)");
 request.withBody(body);
 try {
 ListStructuredLogsWithTimeRangeResponse response =
client.listStructuredLogsWithTimeRange(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

```
}
}
```

## Python

- 当 “query” 参数填写为 select \*

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListStructuredLogsWithTimeRangeRequest()
 request.log_stream_id = "{log_stream_id}"
 timeRangebody = TimeRange(
 sql_time_zone="UTC",
 start_time=1613806109000,
 end_time=1613806110000,
 start_time_gt=False,
 end_time_lt=False
)
 request.body = QueryLtsStructLogParamsNew(
 time_range=timeRangebody,
 format="k-v",
 query="select *"
)
 response = client.list_structured_logs_with_time_range(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 当 “query” 参数填写为 select count(test3) as mycount, sum(test4) as mysum

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
```

```
In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = ListStructuredLogsWithTimeRangeRequest()
 request.log_stream_id = "{log_stream_id}"
 timeRangebody = TimeRange(
 sql_time_zone="UTC",
 start_time=1613806109000,
 end_time=1613806110000,
 start_time_gt=False,
 end_time_lt=False
)
 request.body = QueryLtsStructLogParamsNew(
 time_range=timeRangebody,
 format="k-v",
 query="select count(test3) as mycount, sum(test4) as mysum"
)
 response = client.list_structured_logs_with_time_range(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 当“query”参数填写为 select count(test3), sum(test4) group by(test1)

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListStructuredLogsWithTimeRangeRequest()
 request.log_stream_id = "{log_stream_id}"
 timeRangebody = TimeRange(
 sql_time_zone="UTC",
 start_time=1613806109000,
```

```
 end_time=1613806110000,
 start_time_gt=False,
 end_time_lt=False
)
 request.body = QueryLtsStructLogParamsNew(
 time_range=timeRangebody,
 format="k-v",
 query="select count(test3), sum(test4) group by(test1)"
)
 response = client.list_structured_logs_with_time_range(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 当 “whether\_to\_rows” 参数填写为 true

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListStructuredLogsWithTimeRangeRequest()
 request.log_stream_id = "{log_stream_id}"
 timeRangebody = TimeRange(
 sql_time_zone="UTC",
 start_time=1613806109000,
 end_time=1613806110000,
 start_time_gt=False,
 end_time_lt=False
)
 request.body = QueryLtsStructLogParamsNew(
 whether_to_rows=True,
 time_range=timeRangebody,
 format="k-v",
 query="select count(test3), sum(test4) group by(test1)"
)
 response = client.list_structured_logs_with_time_range(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 当 “query” 参数填写为 select \*

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListStructuredLogsWithTimeRangeRequest{}
 request.LogStreamId = "{log_stream_id}"
 sqlTimeZoneTimeRange := "UTC"
 startTimeGtTimeRange := false
 endTimeLtTimeRange := false
 timeRangebody := &model.TimeRange{
 SqlTimeZone: &sqlTimeZoneTimeRange,
 StartTime: int64(1613806109000),
 EndTime: int64(1613806110000),
 StartTimeGt: &startTimeGtTimeRange,
 EndTimeLt: &endTimeLtTimeRange,
 }
 request.Body = &model.QueryLtsStructLogParamsNew{
 TimeRange: timeRangebody,
 Format: "k-v",
 Query: "select *",
 }
 response, err := client.ListStructuredLogsWithTimeRange(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

- 当 “query” 参数填写为 select count(test3) as mycount, sum(test4) as mysum

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
)
```

```
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{projectId}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build()

 request := &model.ListStructuredLogsWithTimeRangeRequest{}
 request.LogStreamId = "{log_stream_id}"
 sqlTimeZoneTimeRange:= "UTC"
 startTimeGtTimeRange:= false
 endTimeLtTimeRange:= false
 timeRangebody := &model.TimeRange{
 SqlTimeZone: &sqlTimeZoneTimeRange,
 StartTime: int64(1613806109000),
 EndTime: int64(1613806110000),
 StartTimeGt: &startTimeGtTimeRange,
 EndTimeLt: &endTimeLtTimeRange,
 }
 request.Body = &model.QueryLtsStructLogParamsNew{
 TimeRange: timeRangebody,
 Format: "k-v",
 Query: "select count(test3) as mycount, sum(test4) as mysum",
 }
 response, err := client.ListStructuredLogsWithTimeRange(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
}
```

- 当“query”参数填写为 select count(test3), sum(test4) group by(test1)

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
```

```
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.ListStructuredLogsWithTimeRangeRequest{}
request.LogStreamId = "{log_stream_id}"
sqlTimeZoneTimeRange:= "UTC"
startTimeGtTimeRange:= false
endTimeLtTimeRange:= false
timeRangebody := &model.TimeRange{
 SqlTimeZone: &sqlTimeZoneTimeRange,
 StartTime: int64(1613806109000),
 EndTime: int64(1613806110000),
 StartTimeGt: &startTimeGtTimeRange,
 EndTimeLt: &endTimeLtTimeRange,
}
request.Body = &model.QueryLtsStructLogParamsNew{
 TimeRange: timeRangebody,
 Format: "k-v",
 Query: "select count(test3), sum(test4) group by(test1)",
}
response, err := client.ListStructuredLogsWithTimeRange(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

- 当“whether\_to\_rows”参数填写为 true

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
```

```
WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.ListStructuredLogsWithTimeRangeRequest{}
request.LogStreamId = "{log_stream_id}"
sqlTimeZoneTimeRange:= "UTC"
startTimeGtTimeRange:= false
endTimeLtTimeRange:= false
timeRangebody := &model.TimeRange{
 SqlTimeZone: &sqlTimeZoneTimeRange,
 StartTime: int64(1613806109000),
 EndTime: int64(1613806110000),
 StartTimeGt: &startTimeGtTimeRange,
 EndTimeLt: &endTimeLtTimeRange,
}
whetherToRowsQueryLtsStructLogParamsNew:= true
request.Body = &model.QueryLtsStructLogParamsNew{
 WhetherToRows: &whetherToRowsQueryLtsStructLogParamsNew,
 TimeRange: timeRangebody,
 Format: "k-v",
 Query: "select count(test3), sum(test4) group by(test1)",
}
response, err := client.ListStructuredLogsWithTimeRange(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 200 | 请求响应成功。                                                                                       |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。                                                 |
| 401 | AuthFailed。鉴权失败，请确认token后再次请求。                                                                |
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到，但是服务内部出错。                                                |

## 错误码

请参见[错误码](#)。

## 6.4.6 查询日志直方图

### 功能介绍

查询关键词搜索条数该接口用于查询日志在某一时间段内，包含关键词的日志上报情况和日志条数统计结果。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/lts/keyword-count

表 6-167 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                                                                  |
|------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-168 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-169 请求 Body 参数

| 参数         | 是否必选 | 参数类型   | 描述   |
|------------|------|--------|------|
| start_time | 是    | String | 开始时间 |

| 参数            | 是否必选 | 参数类型    | 描述                                                                                                                                                            |
|---------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| end_time      | 是    | String  | 结束时间                                                                                                                                                          |
| step_interval | 是    | Long    | 时间步长，单位为毫秒（ms）。<br>具体请参考如下公式计算：<br>$(end\_time - start\_time) / 1000 * 1000 / 60$ ，其中 $/ 1000 * 1000 /$ 表示取整。<br><b>说明</b><br>如果计算出的时间步长小于等于1000时，则时间步长为1000。 |
| group_id      | 是    | String  | 日志组ID<br>最小长度： <b>36</b><br>最大长度： <b>36</b>                                                                                                                   |
| stream_id     | 是    | String  | 日志流ID<br>最小长度： <b>36</b><br>最大长度： <b>36</b>                                                                                                                   |
| key_word      | 是    | String  | 关键词指相邻两个分词符之间的单词。                                                                                                                                             |
| is_iterative  | 否    | Boolean | 日志迭代查询，默认为false（不开启迭代），true为开启迭代。<br>枚举值：<br>• <b>true</b>                                                                                                    |

## 响应参数

状态码：200

表 6-170 响应 Body 参数

| 参数              | 参数类型    | 描述     |
|-----------------|---------|--------|
| count           | Long    | 日志条数   |
| histogram       | String  | 直方图结果  |
| isQueryComplete | Boolean | 是否查询完成 |

状态码：400

表 6-171 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-172 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

查询日志直方图

```
POST https://{endpoint}/v2/{project_id}/lts/keyword-count
```

```
{
 "group_id" : "00330565-5baf-4e0d-bd16-ba0c6b951d9a",
 "stream_id" : "715cda3b-e17f-492a-a6ca-98a1ba16ad8c",
 "end_time" : 1637820813605,
 "start_time" : 1637817213605,
 "key_word" : "test",
 "step_interval" : 6000
}
```

## 响应示例

状态码：200

查询直方图数据请求响应成功

```
{
 "count" : 1,
 "histogram" : [{
 "num" : 1,
 "startTime" : 1637821594579,
 "endTime" : 1637821595000
 }, {
 "num" : 0,
 "startTime" : 1637821654000,
 "endTime" : 1637821654579
 }]
}
```

**状态码：400**

BadRequest 非法请求建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.0601",
 "error_msg": "must be less than or equal to 86400000"
}
```

**SDK 代码示例**

SDK代码示例如下。

**Java****查询日志直方图**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListLogHistogramSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListLogHistogramRequest request = new ListLogHistogramRequest();
 QueryLogKeyWordCountRequestBody body = new QueryLogKeyWordCountRequestBody();
 body.withKeyword("test");
 body.withStreamId("715cda3b-e17f-492a-a6ca-98a1ba16ad8c");
 body.withGroupId("00330565-5baf-4e0d-bd16-ba0c6b951d9a");
 body.withStepInterval(6000L);
 body.withEndTime("1637820813605");
 body.withStartTime("1637817213605");
 request.withBody(body);
 try {
 ListLogHistogramResponse response = client.listLogHistogram(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 查询日志直方图

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListLogHistogramRequest()
 request.body = QueryLogKeywordCountRequestBody(
 key_word="test",
 stream_id="715cda3b-e17f-492a-a6ca-98a1ba16ad8c",
 group_id="00330565-5baf-4e0d-bd16-ba0c6b951d9a",
 step_interval=6000,
 end_time="1637820813605",
 start_time="1637817213605"
)
 response = client.list_log_histogram(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 查询日志直方图

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
```

```
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build()
)

 request := &model.ListLogHistogramRequest{}
 request.Body = &model.QueryLogKeywordCountRequestBody{
 Keyword: "test",
 StreamId: "715cda3b-e17f-492a-a6ca-98a1ba16ad8c",
 GroupId: "00330565-5baf-4e0d-bd16-ba0c6b951d9a",
 StepInterval: int64(6000),
 EndTime: "1637820813605",
 StartTime: "1637817213605",
 }
 response, err := client.ListLogHistogram(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                         |
|-----|--------------------------------------------|
| 200 | 查询直方图数据请求响应成功                              |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求。       |
| 500 | InternalServerError。表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.4.7 查询上下文日志

### 功能介绍

该接口用于查询指定日志前（上文）后（下文）的若干条日志。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/groups/{log\_group\_id}/streams/{log\_stream\_id}/context

表 6-173 路径参数

| 参数            | 是否必选 | 参数类型   | 描述                                                                                                                                   |
|---------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------|
| project_id    | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：32<br>最大长度：32  |
| log_group_id  | 是    | String | 日志组ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：36<br>最大长度：36 |
| log_stream_id | 是    | String | 日志流ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：36<br>最大长度：36 |

### 请求参数

表 6-174 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                          |
|--------------|------|--------|-------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30 |

表 6-175 请求 Body 参数

| 参数            | 是否必选 | 参数类型    | 描述                                                                             |
|---------------|------|---------|--------------------------------------------------------------------------------|
| line_num      | 否    | String  | 日志单行序列号，字段值需要从日志结果中获取，纳秒级时间戳。                                                  |
| __time__      | 否    | String  | 自定义时间特性时间字段，字段值需要从日志结果中获取，毫秒级时间戳。若已开启云端结构化自定义时间功能，需要使用该字段和line_num字段共同进行上下文查询。 |
| backwardsSize | 否    | Integer | 指定起始日志往前（上文）的日志条数，取值范围 [0, 500]，默认值100<br>最小值：0<br>最大值：500                     |
| forwardsSize  | 否    | Integer | 指定起始日志往后（下文）的日志条数，取值范围 [0, 500]，默认值100<br>最小值：0<br>最大值：500                     |

## 响应参数

状态码：200

表 6-176 响应 Body 参数

| 参数          | 参数类型                                         | 描述                        |
|-------------|----------------------------------------------|---------------------------|
| logs        | Array of <a href="#">LogContents</a> objects | 上下文日志信息。                  |
| total_count | Integer                                      | 返回的总日志条数，包含请求参数中所指定的起始日志。 |

| 参数              | 参数类型    | 描述                              |
|-----------------|---------|---------------------------------|
| backwards_count | Integer | 向前查询到的日志条数。<br>最小值：0<br>最大值：500 |
| forwards_count  | Integer | 向后查询到的日志条数。<br>最小值：0<br>最大值：500 |
| isQueryComplete | Boolean | 是否查询完成。                         |

表 6-177 LogContents

| 参数       | 参数类型                | 描述                               |
|----------|---------------------|----------------------------------|
| content  | String              | 日志原数据。<br>最小长度：1<br>最大长度：10000   |
| line_num | String              | 日志单行序列号。<br>最小长度：19<br>最大长度：19   |
| labels   | Map<String,String > | 该条日志包含的 labels,查询到的日志不同所包含的字段不同。 |

状态码：400

表 6-178 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-179 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

- 查询上下文日志

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/context
```

```
{
 "line_num": "1725154216000000000",
 "backwardsSize": 100,
 "forwardsSize": 100
}
```

- 查询上下文日志，且已开启云端结构化的自定义时间功能。

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/context
```

```
{
 "line_num": "1725154216000000000",
 "_time_": "1725154216000",
 "backwardsSize": 100,
 "forwardsSize": 100
}
```

## 响应示例

状态码：200

查询上下文日志数据请求响应成功

```
{
 "total_count": 3,
 "backwards_count": 1,
 "forwards_count": 2,
 "logs": [{
 "content": "2024-09-01/09:30:00 this log is Error NO 1",
 "labels": {
 "appName": "default_appname",
 "containerName": "CONFIG_FILE",
 "clusterName": "CONFIG_FILE",
 "podName": "default_procname",
 "clusterId": "CONFIG_FILE",
 "nameSpace": "CONFIG_FILE",
 "category": "LTS"
 }
 }, {
 "line_num": "1725154215000000000"
 }, {
 "content": "2024-09-01/09:30:01 this log is Error NO 1",
 "labels": {
 "appName": "default_appname",
 "containerName": "CONFIG_FILE",
 "clusterName": "CONFIG_FILE",
 "podName": "default_procname",
 "clusterId": "CONFIG_FILE",
 "nameSpace": "CONFIG_FILE",

```

```
"category" : "LTS"
},
"line_num" : "1725154216000000000"
}, {
"content" : "2024-09-01/09:30:02 this log is Error NO 1",
"labels" : {
"appName" : "default_appname",
"containerName" : "CONFIG_FILE",
"clusterName" : "CONFIG_FILE",
"podName" : "default_procname",
"clusterId" : "CONFIG_FILE",
"nameSpace" : "CONFIG_FILE",
"category" : "LTS"
},
"line_num" : "1725154217000000000"
}]
}
```

### 状态码: 400

BadRequest 非法请求建议根据error\_msg直接修改该请求。

```
{
"error_code" : "LTS.0009",
"error_msg" : "Failed to validate the request body"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 查询上下文日志

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListLogContextSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
```

```
 .build();
 ListLogContextRequest request = new ListLogContextRequest();
 request.withLogGroupId("{log_group_id}");
 request.withLogStreamId("{log_stream_id}");
 ListLogContextRequestBody body = new ListLogContextRequestBody();
 body.withForwardsSize(100);
 body.withBackwardsSize(100);
 body.withLineNum("1725154216000000000");
 request.withBody(body);
 try {
 ListLogContextResponse response = client.listLogContext(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

- 查询上下文日志，且已开启云端结构化的自定义时间功能。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListLogContextSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListLogContextRequest request = new ListLogContextRequest();
 request.withLogGroupId("{log_group_id}");
 request.withLogStreamId("{log_stream_id}");
 ListLogContextRequestBody body = new ListLogContextRequestBody();
 body.withForwardsSize(100);
 body.withBackwardsSize(100);
 body.withTime("1725154216000");
 body.withLineNum("1725154216000000000");
 request.withBody(body);
```

```
try {
 ListLogContextResponse response = client.listLogContext(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 查询上下文日志

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListLogContextRequest()
 request.log_group_id = "{log_group_id}"
 request.log_stream_id = "{log_stream_id}"
 request.body = ListLogContextRequestBody(
 forwards_size=100,
 backwards_size=100,
 line_num="1725154216000000000"
)
 response = client.list_log_context(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 查询上下文日志，且已开启云端结构化的自定义时间功能。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListLogContextRequest()
 request.log_group_id = "{log_group_id}"
 request.log_stream_id = "{log_stream_id}"
 request.body = ListLogContextRequestBody(
 forwards_size=100,
 backwards_size=100,
 time_="1725154216000",
 line_num="1725154216000000000"
)
 response = client.list_log_context(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 查询上下文日志

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()
```

```
client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.ListLogContextRequest{}
request.LogGroupId = "{log_group_id}"
request.LogStreamId = "{log_stream_id}"
forwardsSizeListLogContextRequestBody:= int32(100)
backwardsSizeListLogContextRequestBody:= int32(100)
lineNumListLogContextRequestBody:= "172515421600000000"
request.Body = &model.ListLogContextRequestBody{
 ForwardsSize: &forwardsSizeListLogContextRequestBody,
 BackwardsSize: &backwardsSizeListLogContextRequestBody,
 LineNum: &lineNumListLogContextRequestBody,
}
response, err := client.ListLogContext(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

- 查询上下文日志，且已开启云端结构化的自定义时间功能。

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListLogContextRequest{}
 request.LogGroupId = "{log_group_id}"
 request.LogStreamId = "{log_stream_id}"
 forwardsSizeListLogContextRequestBody:= int32(100)
 backwardsSizeListLogContextRequestBody:= int32(100)
 timeListLogContextRequestBody:= "1725154216000"
 lineNumListLogContextRequestBody:= "172515421600000000"
 request.Body = &model.ListLogContextRequestBody{
 ForwardsSize: &forwardsSizeListLogContextRequestBody,
 BackwardsSize: &backwardsSizeListLogContextRequestBody,
```

```
 Time: &timeListLogContextRequestBody,
 LineNum: &lineNumListLogContextRequestBody,
 }
 response, err := client.ListLogContext(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                         |
|-----|--------------------------------------------|
| 200 | 查询上下文日志数据请求响应成功                            |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求。       |
| 500 | InternalServerError。表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.4.8 取消收藏

### 功能介绍

取消日志流收藏

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v1.0/{project\_id}/lts/favorite/{fav\_res\_id}

表 6-180 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |
| fav_res_id | 是    | String | 收藏资源id                                                                             |

## 请求参数

表 6-181 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                             |
|--------------|------|--------|--------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1<br>最大长度: 10000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                 |

## 响应参数

状态码: 400

表 6-182 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-183 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-184 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-185 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

取消日志流收藏

```
/v1.0/{project_id}/lts/favorite/{fav_res_id}
```

## 响应示例

状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message": {
 "code": "LTS.0009",
 "details": "update favorite failed"
 }
}
```

```
}
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class DeletefavoriteSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 DeletefavoriteRequest request = new DeletefavoriteRequest();
 request.withFavResId("{fav_res_id}");
 try {
 DeletefavoriteResponse response = client.deletefavorite(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

```
}
}
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = DeletefavoriteRequest()
 request.fav_res_id = "{fav_res_id}"
 response = client.deletefavorite(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()
```

```
client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.DeletefavoriteRequest{}
request.FavResId = "{fav_res_id}"
response, err := client.Deletefavorite(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 200 | 取消收藏成功                              |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.4.9 创建日志收藏

### 功能介绍

创建日志流收藏

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v1.0/{project\_id}/lts/favorite

表 6-186 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-187 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                   |
|--------------|------|--------|------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=utf8。                |

表 6-188 请求 Body 参数

| 参数                     | 是否必选 | 参数类型    | 描述                           |
|------------------------|------|---------|------------------------------|
| eps_id                 | 否    | String  | 企业项目id                       |
| favorite_resource_id   | 是    | String  | 收藏资源id                       |
| favorite_resource_type | 是    | String  | 收藏资源类型                       |
| log_group_id           | 是    | String  | 日志组id                        |
| log_group_name         | 否    | String  | 日志组名称, 日志组名称有误时按照日志组id创建日志收藏 |
| log_stream_id          | 是    | String  | 日志流id                        |
| log_stream_name        | 否    | String  | 日志流名称, 日志流名称有误时按照日志流id创建日志收藏 |
| is_global              | 是    | Boolean | 是否支持全局化, 必填true, 否则创建不了日志收藏  |

## 响应参数

状态码：201

表 6-189 响应 Body 参数

| 参数                     | 参数类型    | 描述                       |
|------------------------|---------|--------------------------|
| create_time            | Integer | 创建时间                     |
| eps_id                 | String  | 企业项目id                   |
| favorite_resource_id   | String  | 收藏资源id                   |
| favorite_resource_type | String  | 收藏资源类型，资源类型为 LOG_STREAM。 |
| log_group_id           | String  | 日志组id                    |
| log_group_name         | String  | 日志组名称                    |
| log_stream_id          | String  | 日志流id                    |
| log_stream_name        | String  | 日志流名称                    |
| project_id             | String  | 项目id                     |
| is_global              | Boolean | 是否开启日志收藏                 |

状态码：400

表 6-190 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-191 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-192 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-193 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

创建日志收藏

```
/v1.0/{project_id}/lts/favorite

{
 "log_group_id": "d91fff37-9d10-47f1-85de-c2840724908f",
 "log_group_name": "lts-group-sgq1",
 "log_stream_id": "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 "log_stream_name": "lts-topic-xxxxtest",
 "eps_id": "0",
 "favorite_resource_id": "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 "favorite_resource_type": "log_stream",
 "is_global": true
}
```

## 响应示例

状态码：201

创建日志收藏成功

```
{
 "create_time": 1669018970929,
 "eps_id": "0",
 "favorite_resource_id": "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 "is_global": true,
 "favorite_resource_type": "LOG_STREAM",
 "log_group_id": "d91fff37-9d10-47f1-85de-c2840724908f",
 "log_group_name": "lts-group-sgq1",
 "log_stream_id": "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 "log_stream_name": "lts-topic-xxxxtest",
 "project_id": "2a473356cca5487f8373be891bffc1cf"
}
```

**状态码：400**

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message" : {
 "code" : "LTS.0603",
 "details" : "group or stream not exist"
 }
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 创建日志收藏

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class CreatefavoriteSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreatefavoriteRequest request = new CreatefavoriteRequest();
 CreatefavoriteReqbody body = new CreatefavoriteReqbody();
 body.withIsGlobal(true);
 body.withLogStreamName("lts-topic-xxxxtest");
```

```
body.withLogStreamId("f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c");
body.withLogGroupName("lts-group-sgq1");
body.withLogGroupId("d91fff37-9d10-47f1-85de-c2840724908f");
body.withFavoriteResourceType("log_stream");
body.withFavoriteResourceId("f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c");
body.withEpsId("0");
request.withBody(body);
try {
 CreatefavoriteResponse response = client.createfavorite(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 创建日志收藏

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreatefavoriteRequest()
 request.body = CreatefavoriteReqbody(
 is_global=True,
 log_stream_name="lts-topic-xxxxtest",
 log_stream_id="f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 log_group_name="lts-group-sgq1",
 log_group_id="d91fff37-9d10-47f1-85de-c2840724908f",
 favorite_resource_type="log_stream",
 favorite_resource_id="f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 eps_id="0"
)
 response = client.createfavorite(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
```

```
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

## Go

### 创建日志收藏

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreatefavoriteRequest{}
 logStreamNameCreatefavoriteReqbody := "lts-topic-xxxxtest"
 logGroupNameCreatefavoriteReqbody := "lts-group-sgq1"
 epsIdCreatefavoriteReqbody := "0"
 request.Body = &model.CreatefavoriteReqbody{
 IsGlobal: true,
 LogStreamName: &logStreamNameCreatefavoriteReqbody,
 LogStreamId: "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 LogGroupName: &logGroupNameCreatefavoriteReqbody,
 LogGroupId: "d91fff37-9d10-47f1-85de-c2840724908f",
 FavoriteResourceType: "log_stream",
 FavoriteResourceId: "f2fb0a2d-d4cd-4bc9-ac12-93c6d255883c",
 EpsId: &epsIdCreatefavoriteReqbody,
 }
 response, err := client.Createfavorite(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 201 | 创建日志收藏成功                            |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.5 日志接入

### 6.5.1 新建跨账号日志接入

#### 功能介绍

新建跨账号日志接入

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v2.0/{project\_id}/lts/createAgencyAccess

表 6-194 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                           |
|------------|------|--------|------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：1<br>最大长度：64 |

## 请求参数

表 6-195 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                             |
|--------------|------|--------|--------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1<br>最大长度: 10000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=utf8。<br>最小长度: 30<br>最大长度: 30                  |

表 6-196 请求 Body 参数

| 参数                  | 是否必选 | 参数类型                                                           | 描述     |
|---------------------|------|----------------------------------------------------------------|--------|
| preview_agency_list | 是    | Array of <a href="#">PreviewAgencyLogAccessReqBody</a> objects | 预览代理列表 |

表 6-197 PreviewAgencyLogAccessReqBody

| 参数                    | 是否必选 | 参数类型   | 描述                                      |
|-----------------------|------|--------|-----------------------------------------|
| agency_access_type    | 是    | String | 日志访问类型<br>枚举值:<br>• <b>AGENCYACCESS</b> |
| agency_log_access     | 是    | String | 跨账号日志接入配置名称                             |
| log_agencyStream_name | 是    | String | 委托日志流名称                                 |
| log_agencyStream_id   | 是    | String | 委托日志流id                                 |
| log_agencyGroup_name  | 是    | String | 委托日志组名称                                 |
| log_agencyGroup_id    | 是    | String | 委托日志组id                                 |

| 参数                          | 是否必选 | 参数类型   | 描述       |
|-----------------------------|------|--------|----------|
| log_beAgency<br>stream_name | 是    | String | 被委托日志流名称 |
| log_beAgency<br>stream_id   | 是    | String | 被委托日志流id |
| log_beAgency<br>group_name  | 是    | String | 被委托日志组名称 |
| log_beAgency<br>group_id    | 是    | String | 被委托日志组id |
| be_agency_pr<br>oject_id    | 是    | String | 被委托项目id  |
| agency_projec<br>t_id       | 是    | String | 委托项目id   |
| agency_domai<br>n_name      | 是    | String | 委托账号名称   |
| agency_name                 | 是    | String | 委托名称     |

## 响应参数

状态码：201

表 6-198 响应 Body 参数

| 参数     | 参数类型                                                            | 描述 |
|--------|-----------------------------------------------------------------|----|
| [数组元素] | Array of<br><b>LTSAccessConfig<br/>InfoRespon200</b><br>objects |    |

表 6-199 LTSAccessConfigInfoRespon200

| 参数                     | 参数类型   | 描述        |
|------------------------|--------|-----------|
| access_config_id       | String | 跨账号日志接入id |
| project_id             | String | 项目ID      |
| access_config_na<br>me | String | 跨账号日志接入名称 |

| 参数                 | 参数类型                                                 | 描述                                                                                                                                                                                            |
|--------------------|------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| access_config_type | Object                                               | 跨账号日志接入类型<br>枚举值：<br><ul style="list-style-type: none"> <li>• Agent</li> <li>• CRD</li> <li>• AGENCYACCESS</li> <li>• K8S_CCE</li> <li>• K8S_CUSTOM</li> <li>• CLOUD_STREAM_AGENCY</li> </ul> |
| group_id           | String                                               | 日志组ID                                                                                                                                                                                         |
| log_group_name     | String                                               | 日志组名称                                                                                                                                                                                         |
| log_stream_id      | String                                               | 日志流ID                                                                                                                                                                                         |
| log_stream_name    | String                                               | 日志流名称                                                                                                                                                                                         |
| create_time        | Long                                                 | 创建时间                                                                                                                                                                                          |
| agency_log_access  | <a href="#">PreviewAgencyLogAccessReqBody</a> object | 委托接入信息                                                                                                                                                                                        |

表 6-200 PreviewAgencyLogAccessReqBody

| 参数                      | 参数类型   | 描述                                                                                 |
|-------------------------|--------|------------------------------------------------------------------------------------|
| agency_access_type      | String | 日志访问类型<br>枚举值：<br><ul style="list-style-type: none"> <li>• AGENCYACCESS</li> </ul> |
| agency_log_access       | String | 跨账号日志接入配置名称                                                                        |
| log_agencyStream_name   | String | 委托日志流名称                                                                            |
| log_agencyStream_id     | String | 委托日志流id                                                                            |
| log_agencyGroup_name    | String | 委托日志组名称                                                                            |
| log_agencyGroup_id      | String | 委托日志组id                                                                            |
| log_beAgencystream_name | String | 被委托日志流名称                                                                           |
| log_beAgencystream_id   | String | 被委托日志流id                                                                           |

| 参数                     | 参数类型   | 描述       |
|------------------------|--------|----------|
| log_beAgencygroup_name | String | 被委托日志组名称 |
| log_beAgencygroup_id   | String | 被委托日志组id |
| be_agency_project_id   | String | 被委托项目id  |
| agency_project_id      | String | 委托项目id   |
| agency_domain_name     | String | 委托账号名称   |
| agency_name            | String | 委托名称     |

状态码：400

表 6-201 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-202 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-203 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-204 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

### 新建跨账号日志接入

POST https://{endpoint}/v2.0/{project\_id}/lts/createAgencyAccess

```
{
 "preview_agency_list": [{
 "agency_log_access": "rule_lb30",
 "agency_access_type": "AGENCYACCESS",
 "agency_name": "wenshufeng",
 "agency_domain_name": "paas_aom_z00418070_01",
 "agency_project_id": "a0a12b069ab4491185d7cf26c3e86ada",
 "be_agency_project_id": "2a473356cca5487f8373be891bffc1cf",
 "log_agencyStream_name": "lts-topic-bug",
 "log_agencyStream_id": "beb169ff-e6e9-4bea-8e77-50afdec74071",
 "log_agencyGroup_name": "lts-group-sgq",
 "log_agencyGroup_id": "f06cbfa0-7243-4031-9380-ae0465bd3997",
 "log_beAgencystream_name": "lts-topic-ECS",
 "log_beAgencystream_id": "36ce06b0-c6bf-436d-9abe-39de86da28bb",
 "log_beAgencygroup_name": "lts-group-sgqECS",
 "log_beAgencygroup_id": "1e749063-d9f5-474f-a537-00cad4e9a108"
 }]
}
```

## 响应示例

### 状态码：201

### 跨账号日志接入成功

```
[{
 "access_config_id": "e929f40e-d1cf-4d59-b656-a2995cbd3229",
 "access_config_name": "rule_lb30",
 "access_config_type": "AGENCYACCESS",
 "agency_log_access": {
 "agency_accessConfig_id": "e929f40e-d1cf-4d59-b656-a2995cbd3229",
 "agency_access_type": "AGENCYACCESS",
 "agency_domain_name": "paas_aom_z00418070_01",
 "agency_log_access": "rule_lb30",
 "agency_name": "wenshufeng",
 "agency_project_id": "a0a12b069ab4491185d7cf26c3e86ada",
 "be_agency_project_id": "2a473356cca5487f8373be891bffc1cf",
 "log_agencyGroup_id": "f06cbfa0-7243-4031-9380-ae0465bd3997",
 "log_agencyGroup_name": "lts-group-sgq",
 "log_agencyStream_id": "beb169ff-e6e9-4bea-8e77-50afdec74071",
 "log_agencyStream_name": "lts-topic-bug",
 "log_beAgencygroup_id": "1e749063-d9f5-474f-a537-00cad4e9a108",
 "log_beAgencygroup_name": "lts-group-sgqECS",
 }
}
```

```
"log_beAgencystream_id" : "36ce06b0-c6bf-436d-9abe-39de86da28bb",
"log_beAgencystream_name" : "lts-topic-ECS"
},
"binary_collect" : false,
"create_time" : 1694400753168,
"group_id" : "1e749063-d9f5-474f-a537-00cad4e9a108",
"hostGroupNum" : 0,
"hostNum" : 0,
"host_group_info_list" : [],
"host_rule_info" : {
 "black_paths" : [],
 "pathType" : "host_file",
 "paths" : [],
 "stderr" : false,
 "stdout" : false
},
"id" : "",
"indexId" : "",
"key" : "",
"log_group_name" : "lts-group-sgqECS",
"log_split" : false,
"log_stream_id" : "36ce06b0-c6bf-436d-9abe-39de86da28bb",
"log_stream_name" : "lts-topic-ECS",
"pathNum" : 0,
"project_id" : "2a473356cca5487f8373be891bffc1cf",
>tag_list" : []
}]
```

**状态码：400**

跨账号日志接入创建失败

```
{
 "message" : {
 "code" : "LTS.0420",
 "details" : "Agency not existed, check domain name and agency name"
 }
}
```

**状态码：500**

服务内部错误

```
{
 "message" : {
 "code" : "LTS.0010",
 "details" : "The system encountered an internal error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

新建跨账号日志接入

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
```

```
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateAgencyAccessSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 CreateAgencyAccessRequest request = new CreateAgencyAccessRequest();
 PreviewAgencyLogAccessReqListBody body = new PreviewAgencyLogAccessReqListBody();
 List<PreviewAgencyLogAccessReqBody> listbodyPreviewAgencyList = new ArrayList<>();
 listbodyPreviewAgencyList.add(
 new PreviewAgencyLogAccessReqBody()
 .withAgencyAccessType(PreviewAgencyLogAccessReqBody.AgencyAccessTypeEnum.fromValue("A
GENCYACCESS"))
 .withAgencyLogAccess("rule_lb30")
 .withLogAgencyStreamName("lts-topic-bug")
 .withLogAgencyStreamId("beb169ff-e6e9-4bea-8e77-50afdec74071")
 .withLogAgencyGroupName("lts-group-sgq")
 .withLogAgencyGroupId("f06cbfa0-7243-4031-9380-ae0465bd3997")
 .withLogBeAgencystreamName("lts-topic-ECS")
 .withLogBeAgencystreamId("36ce06b0-c6bf-436d-9abe-39de86da28bb")
 .withLogBeAgencygroupName("lts-group-sgqECS")
 .withLogBeAgencygroupId("1e749063-d9f5-474f-a537-00cad4e9a108")
 .withBeAgencyProjectId("2a473356cca5487f8373be891bffc1cf")
 .withAgencyProjectId("a0a12b069ab4491185d7cf26c3e86ada")
 .withAgencyDomainName("paas_aom_z00418070_01")
 .withAgencyName("wenshufeng")
);
 body.withPreviewAgencyList(listbodyPreviewAgencyList);
 request.withBody(body);
 try {
 CreateAgencyAccessResponse response = client.createAgencyAccess(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

### 新建跨账号日志接入

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateAgencyAccessRequest()
 listPreviewAgencyListbody = [
 PreviewAgencyLogAccessReqBody(
 agency_access_type="AGENCYACCESS",
 agency_log_access="rule_lb30",
 log_agency_stream_name="lts-topic-bug",
 log_agency_stream_id="beb169ff-e6e9-4bea-8e77-50afdec74071",
 log_agency_group_name="lts-group-sgq",
 log_agency_group_id="f06cbfa0-7243-4031-9380-ae0465bd3997",
 log_be_agencystream_name="lts-topic-ECS",
 log_be_agencystream_id="36ce06b0-c6bf-436d-9abe-39de86da28bb",
 log_be_agencygroup_name="lts-group-sgqECS",
 log_be_agencygroup_id="1e749063-d9f5-474f-a537-00cad4e9a108",
 be_agency_project_id="2a473356cca5487f8373be891bffc1cf",
 agency_project_id="a0a12b069ab4491185d7cf26c3e86ada",
 agency_domain_name="paas_aom_z00418070_01",
 agency_name="wenshufeng"
)
]
 request.body = PreviewAgencyLogAccessReqListBody(
 preview_agency_list=listPreviewAgencyListbody
)
 response = client.create_agency_access(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 新建跨账号日志接入

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)
```

```
func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateAgencyAccessRequest{}
 var listPreviewAgencyListbody = []model.PreviewAgencyLogAccessReqBody{
 {
 AgencyAccessType:
model.GetPreviewAgencyLogAccessReqBodyAgencyAccessTypeEnum().AGENCYACCESS,
 AgencyLogAccess: "rule_lb30",
 LogAgencyStreamName: "lts-topic-bug",
 LogAgencyStreamId: "beb169ff-e6e9-4bea-8e77-50afdec74071",
 LogAgencyGroupName: "lts-group-sgq",
 LogAgencyGroupId: "f06cbfa0-7243-4031-9380-ae0465bd3997",
 LogBeAgencystreamName: "lts-topic-ECS",
 LogBeAgencystreamId: "36ce06b0-c6bf-436d-9abe-39de86da28bb",
 LogBeAgencygroupName: "lts-group-sgqECS",
 LogBeAgencygroupid: "1e749063-d9f5-474f-a537-00cad4e9a108",
 BeAgencyProjectId: "2a473356cca5487f8373be891bffc1cf",
 AgencyProjectId: "a0a12b069ab4491185d7cf26c3e86ada",
 AgencyDomainName: "paas_aom_z00418070_01",
 AgencyName: "wenshufeng",
 },
 }
 request.Body = &model.PreviewAgencyLogAccessReqListBody{
 PreviewAgencyList: listPreviewAgencyListbody,
 }
 response, err := client.CreateAgencyAccess(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述          |
|-----|-------------|
| 201 | 跨账号日志接入成功   |
| 400 | 跨账号日志接入创建失败 |

| 状态码 | 描述     |
|-----|--------|
| 500 | 服务内部错误 |

## 错误码

请参见[错误码](#)。

## 6.5.2 查询日志接入

### 功能介绍

查询日志接入列表

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v3/{project\_id}/lts/access-config-list

表 6-205 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取账号租户ID、项目资源集ID、日志组ID、日志流ID</a> 。<br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-206 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                          |
|--------------|------|--------|-------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30 |

表 6-207 请求 Body 参数

| 参数                      | 是否必选 | 参数类型                                             | 描述                            |
|-------------------------|------|--------------------------------------------------|-------------------------------|
| access_config_name_list | 否    | Array of strings                                 | 接入配置名称列表<br>最小长度：1<br>最大长度：64 |
| host_group_name_list    | 否    | Array of strings                                 | 主机组名称列表<br>最小长度：1<br>最大长度：64  |
| log_group_name_list     | 否    | Array of strings                                 | 日志组名称列表<br>最小长度：1<br>最大长度：64  |
| log_stream_name_list    | 否    | Array of strings                                 | 日志流名称列表<br>最小长度：1<br>最大长度：64  |
| access_config_tag_list  | 否    | Array of <a href="#">accessConfigTag</a> objects | 接入配置标签，KEY不能重复,最多20个标签        |

表 6-208 accessConfigTag

| 参数    | 是否必选 | 参数类型   | 描述                            |
|-------|------|--------|-------------------------------|
| key   | 否    | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | 否    | String | 标签Value<br>最小长度：0<br>最大长度：255 |

## 响应参数

状态码：200

表 6-209 响应 Body 参数

| 参数     | 参数类型                                              | 描述                          |
|--------|---------------------------------------------------|-----------------------------|
| result | Array of <a href="#">AccessConfigInfo</a> objects | 日志接入列表                      |
| total  | Long                                              | 日志接入总数<br>最小值：0<br>最大值：1000 |

表 6-210 AccessConfigInfo

| 参数                   | 参数类型                                               | 描述                                      |
|----------------------|----------------------------------------------------|-----------------------------------------|
| access_config_id     | String                                             | 日志接入ID<br>最小长度：36<br>最大长度：36            |
| access_config_name   | String                                             | 日志接入名称<br>最小长度：1<br>最大长度：64             |
| access_config_type   | String                                             | 日志接入类型。AGENT：主机接入枚举值：<br>• <b>AGENT</b> |
| create_time          | Long                                               | 创建时间<br>最小值：1<br>最大值：9999999999999      |
| access_config_detail | <a href="#">AccessConfigDetailCreate</a> object    | 日志接入详细信息                                |
| log_info             | <a href="#">AccessConfigQueryLogInfo</a> object    | 日志接入日志详情                                |
| host_group_info      | <a href="#">AccessConfigHostGroupIdList</a> object | 日志接入主机组ID列表                             |
| access_config_tag    | Array of <a href="#">accessConfigTag</a> objects   | 标签信息。KEY不能重复,最多20个标签                    |
| log_split            | Boolean                                            | 日志拆分                                    |
| binary_collect       | Boolean                                            | 二进制收集                                   |
| cluster_id           | String                                             | CCE集群ID                                 |

| 参数                  | 参数类型                                    | 描述               |
|---------------------|-----------------------------------------|------------------|
| encoding_format     | String                                  | 编码格式，默认UTF-8     |
| incremental_collect | Boolean                                 | 采集策略：增量/全量       |
| processor_type      | String                                  | IC结构化解析类型        |
| demo_log            | String                                  | 示例日志             |
| demo_fields         | Array of <b>DemoFieldAccess</b> objects | 示例日志解析字段         |
| processors          | Array of <b>Processor</b> objects       | IC结构化解析器         |
| application_id      | String                                  | ServiceStage应用ID |
| environment_id      | String                                  | ServiceStage环境ID |
| component_id        | Array of strings                        | ServiceStage组件ID |

表 6-211 AccessConfigDeatilCreate

| 参数    | 参数类型             | 描述                                                                                                                                                                    |
|-------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| paths | Array of strings | 采集路径。<br>1. 路径必须以/或者字母\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*：不能以/** /*开头<br>4.**只能出现一次``<br>CCE类型中 容器路径和主机路径必填，标准输出不用<br>最小长度：1<br>最大长度：128<br>数组长度：1 - 9 |

| 参数                 | 参数类型                                                    | 描述                                                                                                                                                                       |
|--------------------|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| black_paths        | Array of strings                                        | 采集路径黑名单。<br>1. 路径必须以/或者字母:\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4. **只能出现一次<br>最小长度: 1<br>最大长度: 128<br>数组长度: 0 - 9                          |
| format             | <a href="#">AccessConfigFormatCreate</a> object         | 日志格式。single与multi必须选择一种。                                                                                                                                                 |
| windows_log_info   | <a href="#">AccessConfigWindowsLogInfoCreate</a> object | 日志接入采集Windows事件日志                                                                                                                                                        |
| stdout             | Boolean                                                 | 标准输出开关, 仅CCE接入类型时使用                                                                                                                                                      |
| stderr             | Boolean                                                 | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                                                                |
| pathType           | String                                                  | CCE接入类型, 仅CCE接入类型时使用<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>HOST_FILE</b></li> <li>• <b>CONTAINER_STDOUT</b></li> <li>• <b>CONTAINER_FILE</b></li> </ul> |
| namespaceRegex     | String                                                  | K8s Namespace正则匹配, 仅CCE接入类型时使用                                                                                                                                           |
| podNameRegex       | String                                                  | K8s Pod正则匹配, 仅CCE接入类型时使用                                                                                                                                                 |
| containerNameRegex | String                                                  | K8s 容器名称正则匹配, 仅CCE接入类型时使用                                                                                                                                                |
| includeLabels      | Map<String,String>                                      | 容器 Label白名单, 最多支持创建30个, keyname不支持重名, 仅CCE接入类型时使用                                                                                                                        |
| excludeLabels      | Map<String,String>                                      | 容器 Label黑名单, 最多支持创建30个, keyname不支持重名, 仅CCE接入类型时使用                                                                                                                        |
| includeEnvs        | Map<String,String>                                      | 环境变量白名单, 最多支持创建30个, keyname不支持重名, 仅CCE接入类型时使用                                                                                                                            |

| 参数               | 参数类型               | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| excludeEnvs      | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| logLabels        | Map<String,String> | 容器 Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| logEnvs          | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| includeK8sLabels | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| excludeK8sLabels | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| logK8s           | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| repeat_collect   | Boolean            | 强制重复采集                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| system_fields    | Array of strings   | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置<br/>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |

| 参数                      | 参数类型               | 描述                                                                                                                                            |
|-------------------------|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| custom_key_value        | Map<String,String> | 自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20<br>键的长度限制为128，允许的字符有a-zA-Z0-9_-<br>值的长度限制为1024<br>若修改时传入此字段，将覆盖原有配置 |
| includeLabelsLogical    | String             | 容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                        |
| excludeLabelsLogical    | String             | 容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                        |
| includeK8sLabelsLogical | String             | K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                      |
| excludeK8sLabelsLogical | String             | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                      |
| includeEnvsLogical      | String             | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                                |
| excludeEnvsLogical      | String             | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效                                                                |

表 6-212 AccessConfigFormatCreate

| 参数     | 参数类型                                                  | 描述         |
|--------|-------------------------------------------------------|------------|
| single | <a href="#">AccessConfigFormatSingleCreate</a> object | 日志接入格式单行日志 |
| multi  | <a href="#">AccessConfigFormatMutilCreate</a> object  | 日志接入格式多行日志 |

表 6-213 AccessConfigFormatSingleCreate

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                             |
|-------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | String | 单行日志。system：系统时间，wildcard：时间通配符。<br>枚举值： <ul style="list-style-type: none"><li>• <b>system</b></li><li>• <b>wildcard</b></li></ul>                                                                                                             |
| value | String | 日志时间。当mode为"system"，则填入当前时间戳。当mode为"wildcard"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss<br>最小长度：1<br>最大长度：64 |

表 6-214 AccessConfigFormatMutilCreate

| 参数   | 参数类型   | 描述                                                                                                                          |
|------|--------|-----------------------------------------------------------------------------------------------------------------------------|
| mode | String | 单行日志。time：日志时间，regular：正则模式。<br>枚举值： <ul style="list-style-type: none"><li>• <b>time</b></li><li>• <b>regular</b></li></ul> |

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                                     |
|-------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | String | <p>日志时间。当mode为"regular"，则输入正则表达式当mode为"time"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss</p> <p>最小长度：1<br/>最大长度：64</p> |

表 6-215 AccessConfigWindowsLogInfoCreate

| 参数          | 参数类型                                          | 描述                                                                                                                                                                                                                                                                                  |
|-------------|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | Array of strings                              | <p>采集Windows事件日志类型。Application：应用系统，System：系统，Security：安全，Setup：启动</p> <p>枚举值：</p> <ul style="list-style-type: none"> <li>• <b>Application</b></li> <li>• <b>System</b></li> <li>• <b>Security</b></li> <li>• <b>Setup</b></li> </ul>                                               |
| time_offset | <a href="#">AccessConfigTimeOffset</a> object | 日志接入偏移时间                                                                                                                                                                                                                                                                            |
| event_level | Array of strings                              | <p>事件等级。information：info，warning：告警，error：错误，critical：关键，verbose：冗长</p> <p>数组长度：1 - 5</p> <p>枚举值：</p> <ul style="list-style-type: none"> <li>• <b>information</b></li> <li>• <b>warning</b></li> <li>• <b>error</b></li> <li>• <b>critical</b></li> <li>• <b>verbose</b></li> </ul> |

表 6-216 AccessConfigTimeOffset

| 参数     | 参数类型   | 描述                                                                                                                        |
|--------|--------|---------------------------------------------------------------------------------------------------------------------------|
| offset | Long   | 偏移时间。<br>当"unit"选择"day"时，范围为1~7天。<br>当"unit"选择"hour"时，范围为1~168小时。<br>当"unit"选择"sec"时，范围为1~604800秒。<br>最小值：1<br>最大值：604800 |
| unit   | String | 偏移时间单位。day：天，hour：小时，sec：秒<br>枚举值： <ul style="list-style-type: none"><li>• day</li><li>• hour</li><li>• sec</li></ul>     |

表 6-217 AccessConfigQueryLogInfo

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_group_id    | String | 日志组ID<br>最小长度：36<br>最大长度：36 |
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_group_name  | String | 日志组名称<br>最小长度：1<br>最大长度：128 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：128 |

表 6-218 AccessConfigHostGroupIdList

| 参数                 | 参数类型             | 描述                              |
|--------------------|------------------|---------------------------------|
| host_group_id_list | Array of strings | 主机组ID列表<br>最小长度: 36<br>最大长度: 36 |

表 6-219 accessConfigTag

| 参数    | 参数类型   | 描述                              |
|-------|--------|---------------------------------|
| key   | String | 标签Key<br>最小长度: 1<br>最大长度: 128   |
| value | String | 标签Value<br>最小长度: 0<br>最大长度: 255 |

表 6-220 DemoFieldAccess

| 参数          | 参数类型   | 描述                |
|-------------|--------|-------------------|
| field_name  | String | 字段名称需和keys中字段保持一致 |
| field_value | String | 字段值               |

表 6-221 Processor

| 参数     | 参数类型                 | 描述                                                                                                                                                                                              |
|--------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | String               | 解析器类型 processor_regex 正则解析<br>processor_split_string 分词符<br>processor_json json解析器类型<br>processor_gotime自定义时间类型<br>processor_filter_regex日志过滤<br>processor_drop删除字段类型<br>processor_rename修改字段类型 |
| detail | <b>Detail</b> object | 解析器配置                                                                                                                                                                                           |

表 6-222 Detail

| 参数                         | 参数类型               | 描述                                                |
|----------------------------|--------------------|---------------------------------------------------|
| source_key                 | String             | 自定义时间key字段名称                                      |
| regex                      | String             | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段                      |
| keys                       | Array of strings   | 字段名称列表 与field_name字段保持一致                          |
| multi_line_regex           | String             | 首行正则表达式                                           |
| keep_source                | Boolean            | 是否上传原始日志                                          |
| keep_source_if_parse_error | Boolean            | 是否上传解析失败日志                                        |
| split_sep                  | String             | 分隔符 自定义字符最大长度10，自定义字符串最大长度30                      |
| split_type                 | String             | 分隔符类型：char-自定义字符；special_char-不可见字符；string-自定义字符串 |
| fields                     | Map<String,String> | 添加的字段列表：<字段名称：字段内容>                               |
| drop_keys                  | Array of strings   | 删除的字段列表                                           |
| source_keys                | Array of strings   | 字段重命名源字段名称列表                                      |
| dest_keys                  | Array of strings   | 字段重命名替换的字段名称列表                                    |
| expand_depth               | Integer            | json解析深度，默认为1层最大4层                                |
| expand_connector           | String             | json解析字段名称连接符                                     |
| source_format              | String             | 自定义时间时间格式                                         |
| source_value               | String             | 自定义时间字段value                                      |
| set_time                   | Boolean            | 是否开启自定义时间的开关                                      |
| include                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256，value最大长度为128，key不可以重复      |
| exclude                    | Map<String,String> | 日志过滤黑名单规则 key长度最大为256，value最大长度为128，key不可以重复      |

**状态码：400**

表 6-223 响应 Body 参数

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8     |
| error_msg  | String | 错误描述<br>最小长度：1<br>最大长度：1000 |

**状态码：500**

表 6-224 响应 Body 参数

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8     |
| error_msg  | String | 错误描述<br>最小长度：1<br>最大长度：1000 |

## 请求示例

查询日志接入信息，根据传入的Body体进行过滤。

```
POST https://{endpoint}/v3/{project_id}/lts/access-config-list
{
 "access_config_name_list": ["采xx2", "22x", "2x", "采集Wjxxxx"],
 "host_group_name_list": ["wwxx"],
 "log_group_name_list": ["lts-grxx", "lts-xx", "lts-gxx"],
 "log_stream_name_list": ["lts-topixx", "lts-txx"],
 "access_config_tag_list": [{
 "key": "xxx",
 "value": "xxx"
 }, {
 "key": "xxx1",
 "value": "xxx1"
 }]
}
```

## 响应示例

**状态码：200**

查询日志接入列表请求响应成功

```
{
 "result": [{
```

```
"access_config_detail": {
 "containerNameRegex": "my",
 "excludeEnvs": {
 "h": "8"
 },
 "excludeK8sLabels": {
 "e": "5"
 },
 "excludeLabels": {
 "b": "2"
 },
 "format": {
 "single": {
 "mode": "system",
 "value": "1678969382000"
 }
 },
 "includeEnvs": {
 "g": "7"
 },
 "includeK8sLabels": {
 "d30": "4"
 },
 "includeLabels": {
 "a": "1"
 },
 "logEnvs": {
 "i": "9"
 },
 "logK8s": {
 "f": "6"
 },
 "logLabels": {
 "c": "3"
 },
 "namespaceRegex": "default",
 "pathType": "container_stdout",
 "paths": [],
 "podNameRegex": "abc",
 "stderr": false,
 "stdout": true
},
"access_config_id": "c3152f88-8b06-4f7f-bbbe-129512f49f87",
"access_config_name": "myapinew322",
"access_config_tag": [{
 "key": "my01",
 "value": "001"
}, {
 "key": "my02",
 "value": "002"
}],
"access_config_type": "K8S_CCE",
"binary_collect": false,
"create_time": 1684467787996,
"host_group_info": {
 "host_group_id_list": ["12b0bbd1-4eda-456b-a641-647aa66bdeab"]
},
"log_info": {
 "log_group_id": "9575cb24-290c-478e-a5db-88d6d1dc513b",
 "log_group_name": "my-group",
 "log_stream_id": "3581bee9-8698-476e-a0ba-b0f310ed99cf",
 "log_stream_name": "lts-topic-api"
},
"log_split": false
}, {
 "access_config_detail": {
 "containerNameRegex": "my",
 "excludeEnvs": {
 "h": "8"
```

```
},
"excludeK8sLabels" : {
 "e" : "5"
},
"excludeLabels" : {
 "b" : "2"
},
"format" : {
 "single" : {
 "mode" : "system",
 "value" : "1678969382000"
 }
},
"includeEnvs" : {
 "g" : "7"
},
"includeK8sLabels" : {
 "d10" : "4",
 "d" : "4",
 "d12" : "4",
 "d11" : "4",
 "d14" : "4",
 "d13" : "4",
 "d16" : "4",
 "d15" : "4",
 "d18" : "4",
 "d17" : "4",
 "d1" : "4",
 "d2" : "4",
 "d3" : "4",
 "d4" : "4",
 "d5" : "4",
 "d6" : "4",
 "d7" : "4",
 "d8" : "4",
 "d9" : "4"
},
"includeLabels" : {
 "a" : "1"
},
"logEnvs" : {
 "i" : "9"
},
"logK8s" : {
 "f" : "6"
},
"logLabels" : {
 "c" : "3"
},
"namespaceRegex" : "default",
"pathType" : "container_stdout",
"paths" : [],
"podNameRegex" : "abc",
"stderr" : false,
"stdout" : true
},
"access_config_id" : "550cd738-7b16-4724-9c59-aba61bf16528",
"access_config_name" : "myapinew32",
"access_config_tag" : [{
 "key" : "my01",
 "value" : "001"
}, {
 "key" : "my02",
 "value" : "002"
}],
"access_config_type" : "K8S_CCE",
"binary_collect" : false,
"create_time" : 1684463134956,
"host_group_info" : {
```

```
"host_group_id_list" : ["12b0bbd1-4eda-456b-a641-647aa66bdeab"]
},
"log_info" : {
 "log_group_id" : "9575cb24-290c-478e-a5db-88d6d1dc513b",
 "log_group_name" : "my-group",
 "log_stream_id" : "3581bee9-8698-476e-a0ba-b0f310ed99cf",
 "log_stream_name" : "lts-topic-api"
},
"log_split" : false
}],
"total" : 2
}
```

**状态码：400**

非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.1807",
 "error_msg" : "Invalid access config name"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0010",
 "error_msg" : "The system encountered an internal error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

查询日志接入信息，根据传入的Body体进行过滤。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListAccessConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
```

```
.withAk(ak)
.withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
ListAccessConfigRequest request = new ListAccessConfigRequest();
GetAccessConfigListRequestBody body = new GetAccessConfigListRequestBody();
List<AccessConfigTag> listbodyAccessConfigTagList = new ArrayList<>();
listbodyAccessConfigTagList.add(
 new AccessConfigTag()
 .withKey("xxx")
 .withValue("xxx")
);
listbodyAccessConfigTagList.add(
 new AccessConfigTag()
 .withKey("xxx1")
 .withValue("xxx1")
);
List<String> listbodyLogStreamNameList = new ArrayList<>();
listbodyLogStreamNameList.add("lts-topixx");
listbodyLogStreamNameList.add("lts-txx");
List<String> listbodyLogGroupNameList = new ArrayList<>();
listbodyLogGroupNameList.add("lts-grxx");
listbodyLogGroupNameList.add("lts-xx");
listbodyLogGroupNameList.add("lts-gxx");
List<String> listbodyHostGroupNameList = new ArrayList<>();
listbodyHostGroupNameList.add("wwwx");
List<String> listbodyAccessConfigNameList = new ArrayList<>();
listbodyAccessConfigNameList.add("采xx2");
listbodyAccessConfigNameList.add("22x");
listbodyAccessConfigNameList.add("2x");
listbodyAccessConfigNameList.add("采集Wjxxxx");
body.withAccessConfigTagList(listbodyAccessConfigTagList);
body.withLogStreamNameList(listbodyLogStreamNameList);
body.withLogGroupNameList(listbodyLogGroupNameList);
body.withHostGroupNameList(listbodyHostGroupNameList);
body.withAccessConfigNameList(listbodyAccessConfigNameList);
request.withBody(body);
try {
 ListAccessConfigResponse response = client.listAccessConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

查询日志接入信息，根据传入的Body体进行过滤。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *
```

```
if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListAccessConfigRequest()
 listAccessConfigTagListbody = [
 AccessConfigTag(
 key="xxx",
 value="xxx"
),
 AccessConfigTag(
 key="xxx1",
 value="xxx1"
)
]
 listLogStreamNameListbody = [
 "lts-topixx",
 "lts-txx"
]
 listLogGroupNameListbody = [
 "lts-grxx",
 "lts-xx",
 "lts-gxx"
]
 listHostGroupNameListbody = [
 "wwxx"
]
 listAccessConfigNameListbody = [
 "采xx2",
 "22x",
 "2x",
 "采集Wjxxxx"
]
 request.body = GetAccessConfigListRequestBody(
 access_config_tag_list=listAccessConfigTagListbody,
 log_stream_name_list=listLogStreamNameListbody,
 log_group_name_list=listLogGroupNameListbody,
 host_group_name_list=listHostGroupNameListbody,
 access_config_name_list=listAccessConfigNameListbody
)
 response = client.list_access_config(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

查询日志接入信息，根据传入的Body体进行过滤。

```
package main

import (
```

```
"fmt"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListAccessConfigRequest{}
 keyAccessConfigTagList:= "xxx"
 valueAccessConfigTagList:= "xxx"
 keyAccessConfigTagList1:= "xxx1"
 valueAccessConfigTagList1:= "xxx1"
 var listAccessConfigTagListbody = []model.AccessConfigTag{
 {
 Key: &keyAccessConfigTagList,
 Value: &valueAccessConfigTagList,
 },
 {
 Key: &keyAccessConfigTagList1,
 Value: &valueAccessConfigTagList1,
 },
 }
 var listLogStreamNameListbody = []string{
 "lts-topixx",
 "lts-txx",
 }
 var listLogGroupNameListbody = []string{
 "lts-grxx",
 "lts-xx",
 "lts-gxx",
 }
 var listHostGroupNameListbody = []string{
 "wwxx",
 }
 var listAccessConfigNameListbody = []string{
 "采xx2",
 "22x",
 "2x",
 "采集Wjxxxx",
 }
 request.Body = &model.GetAccessConfigListRequestBody{
 AccessConfigTagList: &listAccessConfigTagListbody,
 LogStreamNameList: &listLogStreamNameListbody,
 LogGroupNameList: &listLogGroupNameListbody,
 HostGroupNameList: &listHostGroupNameListbody,
 AccessConfigNameList: &listAccessConfigNameListbody,
 }
}
```

```
response, err := client.ListAccessConfig(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                         |
|-----|----------------------------|
| 200 | 查询日志接入列表请求响应成功             |
| 400 | 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。     |

## 错误码

请参见[错误码](#)。

## 6.5.3 创建日志接入

### 功能介绍

创建日志接入

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v3/{project\_id}/lts/access-config

表 6-225 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取账号租户ID、项目资源集ID、日志组ID、日志流ID</a> 。<br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-226 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-227 请求 Body 参数

| 参数                   | 是否必选 | 参数类型                                                    | 描述                                                                                                             |
|----------------------|------|---------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| access_config_name   | 是    | String                                                  | 日志接入名称。<br>满足正则表达式： $^(?!)(?!)(?!.*?.$)[\u4e00-\u9fa5a-zA-Z0-9-]{1,64}$$<br>最小长度：1<br>最大长度：64                  |
| access_config_type   | 是    | String                                                  | 日志接入类型。AGENT：ECS接入，K8S_CCE:CCE接入<br>枚举值： <ul style="list-style-type: none"><li>AGENT</li><li>K8S_CCE</li></ul> |
| access_config_detail | 是    | <a href="#">AccessConfigDeatilCreate</a> object         | 访问配置详细信息                                                                                                       |
| log_info             | 是    | <a href="#">AccessConfigBaseLogInfoCreate</a> object    | 日志信息                                                                                                           |
| host_group_info      | 否    | <a href="#">AccessConfigHostGroupdListCreate</a> object | 主机组信息                                                                                                          |
| access_config_tag    | 否    | Array of <a href="#">accessConfigTag</a> objects        | 标签信息。KEY不能重复,最多20个标签                                                                                           |

| 参数                        | 是否必选 | 参数类型                                             | 描述                                                                                                                                                       |
|---------------------------|------|--------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| binary_collect            | 否    | Boolean                                          | 二进制采集                                                                                                                                                    |
| log_split                 | 否    | Boolean                                          | 日志拆分                                                                                                                                                     |
| cluster_id                | 否    | String                                           | 集群ID                                                                                                                                                     |
| incremental_collect       | 否    | Boolean                                          | 是否增量采集 true 为是 false为否（全量采集）                                                                                                                             |
| encoding_format           | 否    | String                                           | 编码格式，支持UTF-8，GBK默认UTF-8                                                                                                                                  |
| processor_type            | 否    | String                                           | IC结构化解析类型包括：<br>SINGLE_LINE 单行全文，<br>MULTI_LINE 多行全文，<br>REGEX 单行正则，<br>MULTI_REGEX 多行正则，<br>SPLIT 分隔符，<br>JSON JSON解析，<br>NGINX nginx解析，<br>COMPOSE组合解析 |
| demo_log                  | 否    | String                                           | 示例日志                                                                                                                                                     |
| demo_fields               | 否    | Array of <a href="#">DemoFieldAccess</a> objects | 示例日志解析字段                                                                                                                                                 |
| processors                | 否    | Array of <a href="#">Processor</a> objects       | IC结构化解析器                                                                                                                                                 |
| application_id            | 否    | String                                           | ServiceStage应用ID                                                                                                                                         |
| environment_id            | 否    | String                                           | ServiceStage环境ID                                                                                                                                         |
| component_id              | 否    | Array of strings                                 | ServiceStage组件ID                                                                                                                                         |
| access_config_type_source | 否    | String                                           | 日志接入自建软件来源                                                                                                                                               |

表 6-228 AccessConfigDeatilCreate

| 参数               | 是否必选 | 参数类型                                    | 描述                                                                                                                                                                         |
|------------------|------|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| paths            | 否    | Array of strings                        | 采集路径。<br>1. 路径必须以/或者字母:\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次`<br>CCE类型中 容器路径和主机路径必填, 标准输出不用<br>最小长度: 1<br>最大长度: 128<br>数组长度: 1 - 9 |
| black_paths      | 否    | Array of strings                        | 采集路径黑名单。<br>1. 路径必须以/或者字母:\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次<br>最小长度: 1<br>最大长度: 128<br>数组长度: 0 - 9                             |
| format           | 是    | AccessConfigFormatCreate object         | 日志格式。single与multi必须选择一种。                                                                                                                                                   |
| windows_log_info | 否    | AccessConfigWindowsLogInfoCreate object | 日志接入采集Windows事件日志                                                                                                                                                          |
| stdout           | 否    | Boolean                                 | 标准输出开关, 仅CCE接入类型时使用                                                                                                                                                        |
| stderr           | 否    | Boolean                                 | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                                                                  |
| pathType         | 否    | String                                  | CCE接入类型, 仅CCE接入类型时使用<br>枚举值:<br><ul style="list-style-type: none"> <li>● HOST_FILE</li> <li>● CONTAINER_STDOUT</li> <li>● CONTAINER_FILE</li> </ul>                        |

| 参数                 | 是否必选 | 参数类型               | 描述                                               |
|--------------------|------|--------------------|--------------------------------------------------|
| namespaceRegex     | 否    | String             | K8s Namespace正则匹配，仅CCE接入类型时使用                    |
| podNameRegex       | 否    | String             | K8s Pod正则匹配，仅CCE接入类型时使用                          |
| containerNameRegex | 否    | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                         |
| includeLabels      | 否    | Map<String,String> | 容器 Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| excludeLabels      | 否    | Map<String,String> | 容器 Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| includeEnvs        | 否    | Map<String,String> | 环境变量白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| excludeEnvs        | 否    | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| logLabels          | 否    | Map<String,String> | 容器 Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logEnvs            | 否    | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用      |
| includeK8sLabels   | 否    | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| excludeK8sLabels   | 否    | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logK8s             | 否    | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用 |
| repeat_collect     | 否    | Boolean            | 强制重复采集                                           |

| 参数                   | 是否必选 | 参数类型               | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------|------|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| system_fields        | 否    | Array of strings   | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置</p> <p>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |
| custom_key_value     | 否    | Map<String,String> | <p>自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20</p> <p>键的长度限制为128，允许的字符有a-zA-Z0-9_-</p> <p>值的长度限制为1024</p> <p>若修改时传入此字段，将覆盖原有配置</p>                                                                                                                                                                                                                                                                                                              |
| includeLabelsLogical | 否    | String             | <p>容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                 |

| 参数                      | 是否必选 | 参数类型   | 描述                                                                                       |
|-------------------------|------|--------|------------------------------------------------------------------------------------------|
| excludeLabelsLogical    | 否    | String | 容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效   |
| includeK8sLabelsLogical | 否    | String | K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| excludeK8sLabelsLogical | 否    | String | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| includeEnvsLogical      | 否    | String | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |
| excludeEnvsLogical      | 否    | String | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |

表 6-229 AccessConfigFormatCreate

| 参数     | 是否必选 | 参数类型                                                  | 描述         |
|--------|------|-------------------------------------------------------|------------|
| single | 否    | <a href="#">AccessConfigFormatSingleCreate</a> object | 日志接入格式单行日志 |
| multi  | 否    | <a href="#">AccessConfigFormatMutilCreate</a> object  | 日志接入格式多行日志 |

表 6-230 AccessConfigFormatSingleCreate

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                               |
|-------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | 否    | String | 单行日志。system: 系统时间, wildcard: 时间通配符。<br>枚举值:<br><ul style="list-style-type: none"> <li>• system</li> <li>• wildcard</li> </ul>                                                                                                                                    |
| value | 否    | String | 日志时间。当mode为”system”, 则填入当前时间戳。当mode为”wildcard”, 则时间通配符: 用日志打印时间来标识一条日志数据, 通过时间通配符来匹配日志, 每条日志的行首显示日志的打印时间; 如果日志中的时间格式为:<br>2019-01-01 23:59:59, 时间通配符应该填写为: YYYY-MM-DD hh:mm:ss; 如果日志中的时间格式为: 19-1-1 23:59:59, 时间通配符应该填写为: YY-M-D hh:mm:ss<br>最小长度: 1<br>最大长度: 64 |

表 6-231 AccessConfigFormatMutilCreate

| 参数   | 是否必选 | 参数类型   | 描述                                                                                                                     |
|------|------|--------|------------------------------------------------------------------------------------------------------------------------|
| mode | 否    | String | 单行日志。time: 日志时间, regular: 正则模式。<br>枚举值:<br><ul style="list-style-type: none"> <li>• time</li> <li>• regular</li> </ul> |

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                                         |
|-------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | 否    | String | <p>日志时间。当mode为"regular", 则输入正则表达式<br/>当mode为"time", 则时间通配符: 用日志打印时间来标识一条日志数据, 通过时间通配符来匹配日志, 每条日志的行首显示日志的打印时间; 如果日志中的时间格式为: 2019-01-01 23:59:59, 时间通配符应该填写为: YYYY-MM-DD hh:mm:ss; 如果日志中的时间格式为: 19-1-1 23:59:59, 时间通配符应该填写为: YY-M-D hh:mm:ss</p> <p>最小长度: 1<br/>最大长度: 64</p> |

表 6-232 AccessConfigWindowsLogInfoCreate

| 参数          | 是否必选 | 参数类型                           | 描述                                                                                                                                                                                                                    |
|-------------|------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | 是    | Array of strings               | <p>采集Windows事件日志类型。<br/>Application: 应用系统, System: 系统, Security: 安全, Setup: 启动</p> <p>枚举值:</p> <ul style="list-style-type: none"> <li>• Application</li> <li>• System</li> <li>• Security</li> <li>• Setup</li> </ul> |
| time_offset | 是    | AccessConfig TimeOffset object | 日志接入偏移时间                                                                                                                                                                                                              |

| 参数          | 是否必选 | 参数类型             | 描述                                                                                                                                                                                                                                             |
|-------------|------|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_level | 是    | Array of strings | 事件等级。information: info, warning: 告警, error: 错误, critical: 关键, verbose: 冗长<br>数组长度: 1 - 5<br>枚举值:<br><ul style="list-style-type: none"> <li>• information</li> <li>• warning</li> <li>• error</li> <li>• critical</li> <li>• verbose</li> </ul> |

表 6-233 AccessConfigTimeOffset

| 参数     | 是否必选 | 参数类型   | 描述                                                                                                                                |
|--------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------|
| offset | 是    | Long   | 偏移时间。<br>当"unit"选择"day"时, 范围为1~7天。<br>当"unit"选择"hour"时, 范围为1~168小时。<br>当"unit"选择"sec"时, 范围为1~604800秒。<br>最小值: 1<br>最大值: 604800    |
| unit   | 是    | String | 偏移时间单位。day: 天, hour: 小时, sec: 秒<br>枚举值:<br><ul style="list-style-type: none"> <li>• day</li> <li>• hour</li> <li>• sec</li> </ul> |

表 6-234 AccessConfigBaseLogInfoCreate

| 参数           | 是否必选 | 参数类型   | 描述                            |
|--------------|------|--------|-------------------------------|
| log_group_id | 是    | String | 日志组ID<br>最小长度: 36<br>最大长度: 36 |

| 参数            | 是否必选 | 参数类型   | 描述                          |
|---------------|------|--------|-----------------------------|
| log_stream_id | 是    | String | 日志流ID<br>最小长度：36<br>最大长度：36 |

表 6-235 AccessConfigHostGroupIdListCreate

| 参数                 | 是否必选 | 参数类型             | 描述                            |
|--------------------|------|------------------|-------------------------------|
| host_group_id_list | 是    | Array of strings | 主机组ID列表<br>最小长度：36<br>最大长度：36 |

表 6-236 accessConfigTag

| 参数    | 是否必选 | 参数类型   | 描述                            |
|-------|------|--------|-------------------------------|
| key   | 否    | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | 否    | String | 标签Value<br>最小长度：0<br>最大长度：255 |

表 6-237 DemoFieldAccess

| 参数          | 是否必选 | 参数类型   | 描述                |
|-------------|------|--------|-------------------|
| field_name  | 否    | String | 字段名称需和keys中字段保持一致 |
| field_value | 否    | String | 字段值               |

表 6-238 Processor

| 参数     | 是否必选 | 参数类型                 | 描述                                                                                                                                                                            |
|--------|------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | 否    | String               | 解析器类型 processor_regex 正则解析 processor_split_string 分词符 processor_json json解析器类型 processor_gotime自定义时间类型 processor_filter_regex日志过滤 processor_drop删除字段类型 processor_rename修改字段类型 |
| detail | 否    | <b>Detail</b> object | 解析器配置                                                                                                                                                                         |

表 6-239 Detail

| 参数                         | 是否必选 | 参数类型               | 描述                                                |
|----------------------------|------|--------------------|---------------------------------------------------|
| source_key                 | 否    | String             | 自定义时间key字段名称                                      |
| regex                      | 否    | String             | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段                      |
| keys                       | 否    | Array of strings   | 字段名称列表 与field_name字段保持一致                          |
| multi_line_regex           | 否    | String             | 首行正则表达式                                           |
| keep_source                | 否    | Boolean            | 是否上传原始日志                                          |
| keep_source_if_parse_error | 否    | Boolean            | 是否上传解析失败日志                                        |
| split_sep                  | 否    | String             | 分隔符 自定义字符最大长度10，自定义字符串最大长度30                      |
| split_type                 | 否    | String             | 分隔符类型：char-自定义字符；special_char-不可见字符；string-自定义字符串 |
| fields                     | 否    | Map<String,String> | 添加的字段列表：<字段名称：字段内容>                               |
| drop_keys                  | 否    | Array of strings   | 删除的字段列表                                           |
| source_keys                | 否    | Array of strings   | 字段重命名源字段名称列表                                      |
| dest_keys                  | 否    | Array of strings   | 字段重命名替换的字段名称列表                                    |

| 参数               | 是否必选 | 参数类型               | 描述                                             |
|------------------|------|--------------------|------------------------------------------------|
| expand_depth     | 否    | Integer            | json解析深度，默认为1层最大4层                             |
| expand_connector | 否    | String             | json解析字段名称连接符                                  |
| source_format    | 否    | String             | 自定义时间时间格式                                      |
| source_value     | 否    | String             | 自定义时间字段value                                   |
| set_time         | 否    | Boolean            | 是否开启自定义时间的开关                                   |
| include          | 否    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复 |
| exclude          | 否    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复 |

## 响应参数

状态码：200

表 6-240 响应 Body 参数

| 参数                   | 参数类型                            | 描述                                 |
|----------------------|---------------------------------|------------------------------------|
| access_config_id     | String                          | 日志接入ID<br>最小长度：36<br>最大长度：36       |
| access_config_name   | String                          | 日志接入名称<br>最小长度：1<br>最大长度：64        |
| access_config_type   | String                          | 日志接入类型。AGENT：主机接入枚举值：<br>• AGENT   |
| create_time          | Long                            | 创建时间<br>最小值：1<br>最大值：9999999999999 |
| access_config_detail | AccessConfigDetailCreate object | 日志接入详细信息                           |
| log_info             | AccessConfigQueryLogInfo object | 日志接入日志详情                           |

| 参数                  | 参数类型                                               | 描述                   |
|---------------------|----------------------------------------------------|----------------------|
| host_group_info     | <a href="#">AccessConfigHostGroupIdList</a> object | 日志接入主机组ID列表          |
| access_config_tag   | Array of <a href="#">accessConfigTag</a> objects   | 标签信息。KEY不能重复,最多20个标签 |
| log_split           | Boolean                                            | 日志拆分                 |
| binary_collect      | Boolean                                            | 二进制收集                |
| cluster_id          | String                                             | CCE集群ID              |
| encoding_format     | String                                             | 编码格式, 默认UTF-8        |
| incremental_collect | Boolean                                            | 采集策略: 增量/全量          |
| processor_type      | String                                             | IC结构化解析类型            |
| demo_log            | String                                             | 示例日志                 |
| demo_fields         | Array of <a href="#">DemoFieldAccess</a> objects   | 示例日志解析字段             |
| processors          | Array of <a href="#">Processor</a> objects         | IC结构化解析器             |
| application_id      | String                                             | ServiceStage应用ID     |
| environment_id      | String                                             | ServiceStage环境ID     |
| component_id        | Array of strings                                   | ServiceStage组件ID     |

表 6-241 AccessConfigDeatilCreate

| 参数               | 参数类型                                                    | 描述                                                                                                                                                                         |
|------------------|---------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| paths            | Array of strings                                        | 采集路径。<br>1. 路径必须以/或者字母:\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次`<br>CCE类型中 容器路径和主机路径必填, 标准输出不用<br>最小长度: 1<br>最大长度: 128<br>数组长度: 1 - 9 |
| black_paths      | Array of strings                                        | 采集路径黑名单。<br>1. 路径必须以/或者字母:\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次<br>最小长度: 1<br>最大长度: 128<br>数组长度: 0 - 9                             |
| format           | <a href="#">AccessConfigFormatCreate</a> object         | 日志格式。single与multi必须选择一种。                                                                                                                                                   |
| windows_log_info | <a href="#">AccessConfigWindowsLogInfoCreate</a> object | 日志接入采集Windows事件日志                                                                                                                                                          |
| stdout           | Boolean                                                 | 标准输出开关, 仅CCE接入类型时使用                                                                                                                                                        |
| stderr           | Boolean                                                 | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                                                                  |
| pathType         | String                                                  | CCE接入类型, 仅CCE接入类型时使用<br>枚举值:<br><ul style="list-style-type: none"><li>• <b>HOST_FILE</b></li><li>• <b>CONTAINER_STDOUT</b></li><li>• <b>CONTAINER_FILE</b></li></ul>       |
| namespaceRegex   | String                                                  | K8s Namespace正则匹配, 仅CCE接入类型时使用                                                                                                                                             |

| 参数                 | 参数类型               | 描述                                               |
|--------------------|--------------------|--------------------------------------------------|
| podNameRegex       | String             | K8s Pod正则匹配，仅CCE接入类型时使用                          |
| containerNameRegex | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                         |
| includeLabels      | Map<String,String> | 容器 Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| excludeLabels      | Map<String,String> | 容器 Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| includeEnvs        | Map<String,String> | 环境变量白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| excludeEnvs        | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| logLabels          | Map<String,String> | 容器 Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logEnvs            | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用      |
| includeK8sLabels   | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| excludeK8sLabels   | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logK8s             | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用 |
| repeat_collect     | Boolean            | 强制重复采集                                           |

| 参数                      | 参数类型                | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| system_fields           | Array of strings    | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置<br/>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |
| custom_key_value        | Map<String,String > | <p>自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20</p> <p>键的长度限制为128，允许的字符有a-zA-Z0-9_-</p> <p>值的长度限制为1024</p> <p>若修改时传入此字段，将覆盖原有配置</p>                                                                                                                                                                                                                                                                                                                     |
| includeLabelsLogical    | String              | <p>容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| excludeLabelsLogical    | String              | <p>容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| includeK8sLabelsLogical | String              | <p>K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                      |

| 参数                      | 参数类型   | 描述                                                                                       |
|-------------------------|--------|------------------------------------------------------------------------------------------|
| excludeK8sLabelsLogical | String | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| includeEnvsLogical      | String | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |
| excludeEnvsLogical      | String | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |

表 6-242 AccessConfigFormatCreate

| 参数     | 参数类型                                                  | 描述         |
|--------|-------------------------------------------------------|------------|
| single | <a href="#">AccessConfigFormatSingleCreate</a> object | 日志接入格式单行日志 |
| multi  | <a href="#">AccessConfigFormatMutilCreate</a> object  | 日志接入格式多行日志 |

表 6-243 AccessConfigFormatSingleCreate

| 参数   | 参数类型   | 描述                                                                                                                                    |
|------|--------|---------------------------------------------------------------------------------------------------------------------------------------|
| mode | String | 单行日志。system：系统时间，wildcard：时间通配符。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>system</b></li> <li>• <b>wildcard</b></li> </ul> |

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                             |
|-------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | String | 日志时间。当mode为"system"，则填入当前时间戳。当mode为"wildcard"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss<br>最小长度：1<br>最大长度：64 |

表 6-244 AccessConfigFormatMutilCreate

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                         |
|-------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | String | 单行日志。time：日志时间，regular：正则模式。<br>枚举值： <ul style="list-style-type: none"><li>time</li><li>regular</li></ul>                                                                                                                                  |
| value | String | 日志时间。当mode为"regular"，则输入正则表达式当mode为"time"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss<br>最小长度：1<br>最大长度：64 |

表 6-245 AccessConfigWindowsLogInfoCreate

| 参数          | 参数类型                                          | 描述                                                                                                                                                                                                                                                                                |
|-------------|-----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | Array of strings                              | 采集Windows事件日志类型。<br>Application: 应用系统, System: 系统, Security: 安全, Setup: 启动<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>Application</b></li> <li>• <b>System</b></li> <li>• <b>Security</b></li> <li>• <b>Setup</b></li> </ul>                                          |
| time_offset | <a href="#">AccessConfigTimeOffset</a> object | 日志接入偏移时间                                                                                                                                                                                                                                                                          |
| event_level | Array of strings                              | 事件等级。information: info, warning: 告警, error: 错误, critical: 关键, verbose: 冗长<br>数组长度: 1 - 5<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>information</b></li> <li>• <b>warning</b></li> <li>• <b>error</b></li> <li>• <b>critical</b></li> <li>• <b>verbose</b></li> </ul> |

表 6-246 AccessConfigTimeOffset

| 参数     | 参数类型 | 描述                                                                                                                                           |
|--------|------|----------------------------------------------------------------------------------------------------------------------------------------------|
| offset | Long | 偏移时间。<br>当"unit"选择"day"时, 范围为1~7天。<br>当"unit"选择"hour"时, 范围为1~168小时。<br>当"unit"选择"sec"时, 范围为1~604800秒。<br>最小值: <b>1</b><br>最大值: <b>604800</b> |

| 参数   | 参数类型   | 描述                                                                                                                                         |
|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------|
| unit | String | 偏移时间单位。day：天，hour：小时，sec：秒<br>枚举值： <ul style="list-style-type: none"><li>• <b>day</b></li><li>• <b>hour</b></li><li>• <b>sec</b></li></ul> |

表 6-247 AccessConfigQueryLogInfo

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_group_id    | String | 日志组ID<br>最小长度：36<br>最大长度：36 |
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_group_name  | String | 日志组名称<br>最小长度：1<br>最大长度：128 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：128 |

表 6-248 AccessConfigHostGroupIdList

| 参数                 | 参数类型             | 描述                            |
|--------------------|------------------|-------------------------------|
| host_group_id_list | Array of strings | 主机组ID列表<br>最小长度：36<br>最大长度：36 |

表 6-249 accessConfigTag

| 参数    | 参数类型   | 描述                            |
|-------|--------|-------------------------------|
| key   | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | String | 标签Value<br>最小长度：0<br>最大长度：255 |

表 6-250 DemoFieldAccess

| 参数          | 参数类型   | 描述                |
|-------------|--------|-------------------|
| field_name  | String | 字段名称需和keys中字段保持一致 |
| field_value | String | 字段值               |

表 6-251 Processor

| 参数     | 参数类型                 | 描述                                                                                                                                                                                              |
|--------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | String               | 解析器类型 processor_regex 正则解析<br>processor_split_string 分词符<br>processor_json json解析器类型<br>processor_gotime自定义时间类型<br>processor_filter_regex日志过滤<br>processor_drop删除字段类型<br>processor_rename修改字段类型 |
| detail | <b>Detail</b> object | 解析器配置                                                                                                                                                                                           |

表 6-252 Detail

| 参数               | 参数类型             | 描述                           |
|------------------|------------------|------------------------------|
| source_key       | String           | 自定义时间key字段名称                 |
| regex            | String           | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段 |
| keys             | Array of strings | 字段名称列表 与field_name字段保持一致     |
| multi_line_regex | String           | 首行正则表达式                      |
| keep_source      | Boolean          | 是否上传原始日志                     |

| 参数                         | 参数类型               | 描述                                                   |
|----------------------------|--------------------|------------------------------------------------------|
| keep_source_if_parse_error | Boolean            | 是否上传解析失败日志                                           |
| split_sep                  | String             | 分隔符 自定义字符最大长度10, 自定义字符串最大长度30                        |
| split_type                 | String             | 分隔符类型: char-自定义字符; special_char-不可见字符; string-自定义字符串 |
| fields                     | Map<String,String> | 添加的字段列表: <字段名称 : 字段内容>                               |
| drop_keys                  | Array of strings   | 删除的字段列表                                              |
| source_keys                | Array of strings   | 字段重命名源字段名称列表                                         |
| dest_keys                  | Array of strings   | 字段重命名替换的字段名称列表                                       |
| expand_depth               | Integer            | json解析深度, 默认为1层最大4层                                  |
| expand_connector           | String             | json解析字段名称连接符                                        |
| source_format              | String             | 自定义时间格式                                              |
| source_value               | String             | 自定义时间字段value                                         |
| set_time                   | Boolean            | 是否开启自定义时间的开关                                         |
| include                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复       |
| exclude                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复       |

状态码: 400

表 6-253 响应 Body 参数

| 参数         | 参数类型   | 描述                            |
|------------|--------|-------------------------------|
| error_code | String | 错误码<br>最小长度: 8<br>最大长度: 8     |
| error_msg  | String | 错误描述<br>最小长度: 1<br>最大长度: 1000 |

**状态码：500****表 6-254 响应 Body 参数**

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8     |
| error_msg  | String | 错误描述<br>最小长度：1<br>最大长度：1000 |

**请求示例**

- 创建日志接入（CCE接入）

POST https://{endpoint}/v3/{project\_id}/lts/access-config

```
{
 "access_config_name": "myapinew322",
 "access_config_type": "K8S_CCE",
 "access_config_detail": {
 "pathType": "CONTAINER_STDOUT",
 "stdout": "true",
 "stderr": "false",
 "format": {
 "single": {
 "mode": "system",
 "value": "1678969382000"
 }
 }
 },
 "namespaceRegex": "default",
 "podNameRegex": "abc",
 "containerNameRegex": "my",
 "includeLabels": {
 "a": "1"
 },
 "excludeLabels": {
 "b": "2"
 },
 "logLabels": {
 "c": "3"
 },
 "includeK8sLabels": {
 "d": "4"
 },
 "excludeK8sLabels": {
 "e": "5"
 },
 "logK8s": {
 "f": "6"
 },
 "includeEnvs": {
 "g": "7"
 },
 "excludeEnvs": {
 "h": "8"
 },
 "logEnvs": {
 "i": "9"
 }
}
```

```
}
},
"log_info" : {
 "log_group_id" : "9575cb24-290c-478e-a5db-88d6d1dc513b",
 "log_stream_id" : "3581bee9-8698-476e-a0ba-b0f310ed99cf"
},
"host_group_info" : {
 "host_group_id_list" : ["12b0bbd1-4eda-456b-a641-647aa66bdeab"]
},
"access_config_tag" : [{
 "key" : "my01",
 "value" : "001"
}, {
 "key" : "my02",
 "value" : "002"
}],
"binary_collect" : "false",
"log_split" : "false"
}
```

- 创建日志接入（ECS接入）

POST https://{endpoint}/v3/{project\_id}/lts/access-config

```
{
 "access_config_name" : "Tesxxx",
 "access_config_type" : "AGENT",
 "access_config_detail" : {
 "paths" : ["/test/xxx", "/texxx"],
 "black_paths" : ["/testxxx", "/tesxxx"],
 "format" : {
 "multi" : {
 "mode" : "time",
 "value" : "YYYY-MM-DD hh:mm:ss"
 }
 },
 "windows_log_info" : {
 "categorys" : ["System", "Security", "Setup"],
 "event_level" : ["warning", "error", "critical", "verbose"],
 "time_offset" : {
 "offset" : 111,
 "unit" : "hour"
 }
 }
 },
 "log_info" : {
 "log_group_id" : "b179326d-c3be-4217-a3d9-xxxx",
 "log_stream_id" : "020a6fa0-4740-4888-af06-98xxxxxx"
 },
 "host_group_info" : {
 "host_group_id_list" : ["4ee44d4f-a72b-40cf-a3c7-1xxxx"]
 },
 "access_config_tag" : [{
 "key" : "xxx",
 "value" : "xxx"
 }, {
 "key" : "xxx1",
 "value" : "xxx1"
 }]
}
```

## 响应示例

状态码：200

创建日志接入请求响应成功

```
{
 "access_config_detail" : {
```

```
"containerNameRegex" : "container-1",
"format" : {
 "single" : {
 "mode" : "system",
 "value" : "1678969382000"
 }
},
"namespaceRegex" : "default",
"pathType" : "container_stdout",
"paths" : [],
"podNameRegex" : "mystdout-6d7458d77c-rhjcc",
"stderr" : true,
"stdout" : true
},
"access_config_id" : "03b16999-95cf-453b-9668-7aa1fafa564e",
"access_config_name" : "myapinew32Y",
"access_config_tag" : [{
 "key" : "my01",
 "value" : "001"
}, {
 "key" : "my02",
 "value" : "002"
}],
"access_config_type" : "K8S_CCE",
"binary_collect" : true,
"create_time" : 1685626665176,
"log_info" : {
 "log_group_id" : "9575cb24-290c-478e-a5db-88d6d1dc513b",
 "log_group_name" : "my-group",
 "log_stream_id" : "eea03c27-e041-4bec-bd03-6afa10a6561a",
 "log_stream_name" : "lts-topic-cceapi"
},
"log_split" : true
}
```

**状态码：400**

非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.1807",
 "error_msg" : "Invalid access config name"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0010",
 "error_msg" : "The system encountered an internal error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 创建日志接入（CCE接入）

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateAccessConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateAccessConfigRequest request = new CreateAccessConfigRequest();
 CreateAccessConfigRequestBody body = new CreateAccessConfigRequestBody();
 List<AccessConfigTag> listbodyAccessConfigTag = new ArrayList<>();
 listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("my01")
 .withValue("001")
);
 listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("my02")
 .withValue("002")
);
 List<String> listHostGroupInfoHostGroupIDList = new ArrayList<>();
 listHostGroupInfoHostGroupIDList.add("12b0bbd1-4eda-456b-a641-647aa66bdeab");
 AccessConfigHostGroupIDListCreate hostGroupInfobody = new
 AccessConfigHostGroupIDListCreate();
 hostGroupInfobody.withHostGroupIDList(listHostGroupInfoHostGroupIDList);
 AccessConfigBaseLogInfoCreate logInfobody = new AccessConfigBaseLogInfoCreate();
 logInfobody.withLogGroupID("9575cb24-290c-478e-a5db-88d6d1dc513b")
 .withLogStreamID("3581bee9-8698-476e-a0ba-b0f310ed99cf");
 Map<String, String> listAccessConfigDetailLogK8s = new HashMap<>();
 listAccessConfigDetailLogK8s.put("f", "6");
 Map<String, String> listAccessConfigDetailExcludeK8sLabels = new HashMap<>();
 listAccessConfigDetailExcludeK8sLabels.put("e", "5");
 Map<String, String> listAccessConfigDetailIncludeK8sLabels = new HashMap<>();
 listAccessConfigDetailIncludeK8sLabels.put("d", "4");
 Map<String, String> listAccessConfigDetailLogEnvs = new HashMap<>();
 listAccessConfigDetailLogEnvs.put("i", "9");
 Map<String, String> listAccessConfigDetailLogLabels = new HashMap<>();
 listAccessConfigDetailLogLabels.put("c", "3");
 Map<String, String> listAccessConfigDetailExcludeEnvs = new HashMap<>();
 listAccessConfigDetailExcludeEnvs.put("h", "8");
 Map<String, String> listAccessConfigDetailIncludeEnvs = new HashMap<>();
 listAccessConfigDetailIncludeEnvs.put("g", "7");
 Map<String, String> listAccessConfigDetailExcludeLabels = new HashMap<>();
 listAccessConfigDetailExcludeLabels.put("b", "2");
 Map<String, String> listAccessConfigDetailIncludeLabels = new HashMap<>();
```

```
listAccessConfigDetailIncludeLabels.put("a", "1");
AccessConfigFormatSingleCreate singleFormat = new AccessConfigFormatSingleCreate();
singleFormat.withMode(AccessConfigFormatSingleCreate.ModeEnum.fromValue("system"))
 .withValue("1678969382000");
AccessConfigFormatCreate formatAccessConfigDetail = new AccessConfigFormatCreate();
formatAccessConfigDetail.withSingle(singleFormat);
AccessConfigDeatilCreate accessConfigDetailbody = new AccessConfigDeatilCreate();
accessConfigDetailbody.withFormat(formatAccessConfigDetail)
 .withStdout(true)
 .withStderr(false)
 .withPathType(AccessConfigDeatilCreate.PathTypeEnum.fromValue("CONTAINER_STDOUT"))
 .withNamespaceRegex("default")
 .withPodNameRegex("abc")
 .withContainerNameRegex("my")
 .withIncludeLabels(listAccessConfigDetailIncludeLabels)
 .withExcludeLabels(listAccessConfigDetailExcludeLabels)
 .withIncludeEnvs(listAccessConfigDetailIncludeEnvs)
 .withExcludeEnvs(listAccessConfigDetailExcludeEnvs)
 .withLogLabels(listAccessConfigDetailLogLabels)
 .withLogEnvs(listAccessConfigDetailLogEnvs)
 .withIncludeK8sLabels(listAccessConfigDetailIncludeK8sLabels)
 .withExcludeK8sLabels(listAccessConfigDetailExcludeK8sLabels)
 .withLogK8s(listAccessConfigDetailLogK8s);
body.withLogSplit(false);
body.withBinaryCollect(false);
body.withAccessConfigTag(listbodyAccessConfigTag);
body.withHostGroupInfo(hostGroupInfobody);
body.withLogInfo(logInfobody);
body.withAccessConfigDetail(accessConfigDetailbody);

body.withAccessConfigType(CreateAccessConfigRequestBody.AccessConfigTypeEnum.fromValue("K8S_CE"));
body.withAccessConfigName("myapinew322");
request.withBody(body);
try {
 CreateAccessConfigResponse response = client.createAccessConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- 创建日志接入（ECS接入）

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateAccessConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD\_SDK\_AK and CLOUD\_SDK\_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
CreateAccessConfigRequest request = new CreateAccessConfigRequest();
CreateAccessConfigRequestBody body = new CreateAccessConfigRequestBody();
List<AccessConfigTag> listbodyAccessConfigTag = new ArrayList<>();
listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("xxx")
 .withValue("xxx")
);
listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("xxx1")
 .withValue("xxx1")
);
List<String> listHostGroupInfoHostGroupIdList = new ArrayList<>();
listHostGroupInfoHostGroupIdList.add("4ee44d4f-a72b-40cf-a3c7-1xxxx");
AccessConfigHostGroupIdListCreate hostGroupInfobody = new
AccessConfigHostGroupIdListCreate();
hostGroupInfobody.withHostGroupIdList(listHostGroupInfoHostGroupIdList);
AccessConfigBaseLogInfoCreate logInfobody = new AccessConfigBaseLogInfoCreate();
logInfobody.withLogGroupId("b179326d-c3be-4217-a3d9-xxxx")
 .withLogStreamId("020a6fa0-4740-4888-af06-98xxxxxx");
List<AccessConfigWindowsLogInfoCreate.EventLevelEnum> listWindowsLogInfoEventLevel = new
ArrayList<>();

listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoCreate.EventLevelEnum.fromValue("w
arning"));

listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoCreate.EventLevelEnum.fromValue("e
rror"));

listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoCreate.EventLevelEnum.fromValue("cr
itical"));

listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoCreate.EventLevelEnum.fromValue("v
erbose"));
AccessConfigTimeOffset timeOffsetWindowsLogInfo = new AccessConfigTimeOffset();
timeOffsetWindowsLogInfo.withOffset(111L)
 .withUnit(AccessConfigTimeOffset.UnitEnum.fromValue("hour"));
List<AccessConfigWindowsLogInfoCreate.CategoriesEnum> listWindowsLogInfoCategories = new
ArrayList<>();

listWindowsLogInfoCategories.add(AccessConfigWindowsLogInfoCreate.CategoriesEnum.fromValue("Sys
tem"));

listWindowsLogInfoCategories.add(AccessConfigWindowsLogInfoCreate.CategoriesEnum.fromValue("Sec
urity"));

listWindowsLogInfoCategories.add(AccessConfigWindowsLogInfoCreate.CategoriesEnum.fromValue("Set
up"));
AccessConfigWindowsLogInfoCreate windowsLogInfoAccessConfigDetail = new
AccessConfigWindowsLogInfoCreate();
```

```
windowsLogInfoAccessConfigDetail.withCategorys(listWindowsLogInfoCategorys)
 .withTimeOffset(timeOffsetWindowsLogInfo)
 .withEventLevel(listWindowsLogInfoEventLevel);
AccessConfigFormatMutilCreate multiFormat = new AccessConfigFormatMutilCreate();
multiFormat.withMode(AccessConfigFormatMutilCreate.ModeEnum.fromValue("time"))
 .withValue("YYYY-MM-DD hh:mm:ss");
AccessConfigFormatCreate formatAccessConfigDetail = new AccessConfigFormatCreate();
formatAccessConfigDetail.withMulti(multiFormat);
List<String> listAccessConfigDetailBlackPaths = new ArrayList<>();
listAccessConfigDetailBlackPaths.add("/testxxx");
listAccessConfigDetailBlackPaths.add("/tesxxx");
List<String> listAccessConfigDetailPaths = new ArrayList<>();
listAccessConfigDetailPaths.add("/test/xxx");
listAccessConfigDetailPaths.add("/texxx");
AccessConfigDeatilCreate accessConfigDetailbody = new AccessConfigDeatilCreate();
accessConfigDetailbody.withPaths(listAccessConfigDetailPaths)
 .withBlackPaths(listAccessConfigDetailBlackPaths)
 .withFormat(formatAccessConfigDetail)
 .withWindowsLogInfo(windowsLogInfoAccessConfigDetail);
body.withAccessConfigTag(listbodyAccessConfigTag);
body.withHostGroupInfo(hostGroupInfobody);
body.withLogInfo(logInfobody);
body.withAccessConfigDetail(accessConfigDetailbody);

body.withAccessConfigType(CreateAccessConfigRequestBody.AccessConfigTypeEnum.fromValue("AGENT"));
body.withAccessConfigName("Tesxxx");
request.withBody(body);
try {
 CreateAccessConfigResponse response = client.createAccessConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 创建日志接入（CCE接入）

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)
```

```
client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = CreateAccessConfigRequest()
 listAccessConfigTagbody = [
 AccessConfigTag(
 key="my01",
 value="001"
),
 AccessConfigTag(
 key="my02",
 value="002"
)
]
 listHostGroupIdListHostGroupInfo = [
 "12b0bbd1-4eda-456b-a641-647aa66bdeab"
]
 hostGroupInfobody = AccessConfigHostGroupIdListCreate(
 host_group_id_list=listHostGroupIdListHostGroupInfo
)
 logInfobody = AccessConfigBaseLogInfoCreate(
 log_group_id="9575cb24-290c-478e-a5db-88d6d1dc513b",
 log_stream_id="3581bee9-8698-476e-a0ba-b0f310ed99cf"
)
 listLogK8sAccessConfigDetail = {
 "f": "6"
 }
 listExcludeK8sLabelsAccessConfigDetail = {
 "e": "5"
 }
 listIncludeK8sLabelsAccessConfigDetail = {
 "d": "4"
 }
 listLogEnvsAccessConfigDetail = {
 "i": "9"
 }
 listLogLabelsAccessConfigDetail = {
 "c": "3"
 }
 listExcludeEnvsAccessConfigDetail = {
 "h": "8"
 }
 listIncludeEnvsAccessConfigDetail = {
 "g": "7"
 }
 listExcludeLabelsAccessConfigDetail = {
 "b": "2"
 }
 listIncludeLabelsAccessConfigDetail = {
 "a": "1"
 }
 singleFormat = AccessConfigFormatSingleCreate(
 mode="system",
 value="1678969382000"
)
 formatAccessConfigDetail = AccessConfigFormatCreate(
 single=singleFormat
)
 accessConfigDetailbody = AccessConfigDetailCreate(
 format=formatAccessConfigDetail,
 stdout=True,
 stderr=False,
 path_type="CONTAINER_STDOUT",
 namespace_regex="default",
 pod_name_regex="abc",
 container_name_regex="my",
```

```
include_labels=listIncludeLabelsAccessConfigDetail,
exclude_labels=listExcludeLabelsAccessConfigDetail,
include_envs=listIncludeEnvsAccessConfigDetail,
exclude_envs=listExcludeEnvsAccessConfigDetail,
log_labels=listLogLabelsAccessConfigDetail,
log_envs=listLogEnvsAccessConfigDetail,
include_k8s_labels=listIncludeK8sLabelsAccessConfigDetail,
exclude_k8s_labels=listExcludeK8sLabelsAccessConfigDetail,
log_k8s=listLogK8sAccessConfigDetail
)
request.body = CreateAccessConfigRequestBody(
 log_split=False,
 binary_collect=False,
 access_config_tag=listAccessConfigTagbody,
 host_group_info=hostGroupInfobody,
 log_info=logInfobody,
 access_config_detail=accessConfigDetailbody,
 access_config_type="K8S_CCE",
 access_config_name="myapinew322"
)
response = client.create_access_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 创建日志接入 (ECS接入)

```
coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateAccessConfigRequest()
 listAccessConfigTagbody = [
 AccessConfigTag(
 key="xxx",
 value="xxx"
),
 AccessConfigTag(
 key="xxx1",
 value="xxx1"
)
]
 listHostGroupIdListHostGroupInfo = [
 "4ee44d4f-a72b-40cf-a3c7-1xxxxx"
]
]
```

```
hostGroupInfobody = AccessConfigHostGroupIdListCreate(
 host_group_id_list=listHostGroupIdListHostGroupInfo
)
logInfobody = AccessConfigBaseLogInfoCreate(
 log_group_id="b179326d-c3be-4217-a3d9-xxxx",
 log_stream_id="020a6fa0-4740-4888-af06-98xxxxxx"
)
listEventLevelWindowsLogInfo = [
 "warning",
 "error",
 "critical",
 "verbose"
]
timeOffsetWindowsLogInfo = AccessConfigTimeOffset(
 offset=111,
 unit="hour"
)
listCategorysWindowsLogInfo = [
 "System",
 "Security",
 "Setup"
]
windowsLogInfoAccessConfigDetail = AccessConfigWindowsLogInfoCreate(
 categorys=listCategorysWindowsLogInfo,
 time_offset=timeOffsetWindowsLogInfo,
 event_level=listEventLevelWindowsLogInfo
)
multiFormat = AccessConfigFormatMutilCreate(
 mode="time",
 value="YYYY-MM-DD hh:mm:ss"
)
formatAccessConfigDetail = AccessConfigFormatCreate(
 multi=multiFormat
)
listBlackPathsAccessConfigDetail = [
 "/testxxx",
 "/tesxxx"
]
listPathsAccessConfigDetail = [
 "/test/xxx",
 "/texxx"
]
accessConfigDetailbody = AccessConfigDeatilCreate(
 paths=listPathsAccessConfigDetail,
 black_paths=listBlackPathsAccessConfigDetail,
 format=formatAccessConfigDetail,
 windows_log_info=windowsLogInfoAccessConfigDetail
)
request.body = CreateAccessConfigRequestBody(
 access_config_tag=listAccessConfigTagbody,
 host_group_info=hostGroupInfobody,
 log_info=logInfobody,
 access_config_detail=accessConfigDetailbody,
 access_config_type="AGENT",
 access_config_name="Tesxxx"
)
response = client.create_access_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 创建日志接入（CCE接入）

```
package main
```

```
import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateAccessConfigRequest{}
 keyAccessConfigTag := "my01"
 valueAccessConfigTag := "001"
 keyAccessConfigTag1 := "my02"
 valueAccessConfigTag1 := "002"
 var listAccessConfigTagbody = []model.AccessConfigTag{
 {
 Key: &keyAccessConfigTag,
 Value: &valueAccessConfigTag,
 },
 {
 Key: &keyAccessConfigTag1,
 Value: &valueAccessConfigTag1,
 },
 }
 var listHostGroupIdListHostGroupInfo = []string{
 "12b0bbd1-4eda-456b-a641-647aa66bdeab",
 }
 hostGroupInfobody := &model.AccessConfigHostGroupIdListCreate{
 HostGroupIdList: listHostGroupIdListHostGroupInfo,
 }
 logInfobody := &model.AccessConfigBaseLogInfoCreate{
 LogGroupId: "9575cb24-290c-478e-a5db-88d6d1dc513b",
 LogStreamId: "3581bee9-8698-476e-a0ba-b0f310ed99cf",
 }
 var listLogK8sAccessConfigDetail = map[string]string{
 "f": "6",
 }
 var listExcludeK8sLabelsAccessConfigDetail = map[string]string{
 "e": "5",
 }
 var listIncludeK8sLabelsAccessConfigDetail = map[string]string{
 "d": "4",
 }
 var listLogEnvsAccessConfigDetail = map[string]string{
 "i": "9",
 }
 var listLogLabelsAccessConfigDetail = map[string]string{
```

```
 "c": "3",
 }
 var listExcludeEnvsAccessConfigDetail = map[string]string{
 "h": "8",
 }
 var listIncludeEnvsAccessConfigDetail = map[string]string{
 "g": "7",
 }
 var listExcludeLabelsAccessConfigDetail = map[string]string{
 "b": "2",
 }
 var listIncludeLabelsAccessConfigDetail = map[string]string{
 "a": "1",
 }
 modeSingle:= model.GetAccessConfigFormatSingleCreateModeEnum().SYSTEM
 valueSingle:= "1678969382000"
 singleFormat := &model.AccessConfigFormatSingleCreate{
 Mode: &modeSingle,
 Value: &valueSingle,
 }
 formatAccessConfigDetail := &model.AccessConfigFormatCreate{
 Single: singleFormat,
 }
 stdoutAccessConfigDetail:= true
 stderrAccessConfigDetail:= false
 pathTypeAccessConfigDetail:=
model.GetAccessConfigDeatilCreatePathTypeEnum().CONTAINER_STDOUT
 namespaceRegexAccessConfigDetail:= "default"
 podNameRegexAccessConfigDetail:= "abc"
 containerNameRegexAccessConfigDetail:= "my"
 accessConfigDetailbody := &model.AccessConfigDeatilCreate{
 Format: formatAccessConfigDetail,
 Stdout: &stdoutAccessConfigDetail,
 Stderr: &stderrAccessConfigDetail,
 PathType: &pathTypeAccessConfigDetail,
 NamespaceRegex: &namespaceRegexAccessConfigDetail,
 PodNameRegex: &podNameRegexAccessConfigDetail,
 ContainerNameRegex: &containerNameRegexAccessConfigDetail,
 IncludeLabels: listIncludeLabelsAccessConfigDetail,
 ExcludeLabels: listExcludeLabelsAccessConfigDetail,
 IncludeEnvs: listIncludeEnvsAccessConfigDetail,
 ExcludeEnvs: listExcludeEnvsAccessConfigDetail,
 LogLabels: listLogLabelsAccessConfigDetail,
 LogEnvs: listLogEnvsAccessConfigDetail,
 IncludeK8sLabels: listIncludeK8sLabelsAccessConfigDetail,
 ExcludeK8sLabels: listExcludeK8sLabelsAccessConfigDetail,
 LogK8s: listLogK8sAccessConfigDetail,
 }
 logSplitCreateAccessConfigRequestBody:= false
 binaryCollectCreateAccessConfigRequestBody:= false
 request.Body = &model.CreateAccessConfigRequestBody{
 LogSplit: &logSplitCreateAccessConfigRequestBody,
 BinaryCollect: &binaryCollectCreateAccessConfigRequestBody,
 AccessConfigTag: &listAccessConfigTagbody,
 HostGroupInfo: hostGroupInfobody,
 LogInfo: logInfobody,
 AccessConfigDetail: accessConfigDetailbody,
 AccessConfigType: model.GetCreateAccessConfigRequestBodyAccessConfigTypeEnum().K8_S_CCE,
 AccessConfigName: "myapinew322",
 }
 response, err := client.CreateAccessConfig(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

- 创建日志接入 (ECS接入)

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateAccessConfigRequest{
 keyAccessConfigTag:= "xxx"
 valueAccessConfigTag:= "xxx"
 keyAccessConfigTag1:= "xxx1"
 valueAccessConfigTag1:= "xxx1"
 var listAccessConfigTagbody = []model.AccessConfigTag{
 {
 Key: &keyAccessConfigTag,
 Value: &valueAccessConfigTag,
 },
 {
 Key: &keyAccessConfigTag1,
 Value: &valueAccessConfigTag1,
 },
 }
 var listHostGroupIdListHostGroupInfo = []string{
 "4ee44d4f-a72b-40cf-a3c7-1xxxxx",
 }
 hostGroupInfobody := &model.AccessConfigHostGroupIdListCreate{
 HostGroupIdList: listHostGroupIdListHostGroupInfo,
 }
 logInfobody := &model.AccessConfigBaseLogInfoCreate{
 LogGroupId: "b179326d-c3be-4217-a3d9-xxxx",
 LogStreamId: "020a6fa0-4740-4888-af06-98xxxxxx",
 }
 var listEventLevelWindowsLogInfo = []model.AccessConfigWindowsLogInfoCreateEventLevel{
 model.GetAccessConfigWindowsLogInfoCreateEventLevelEnum().WARNING,
 model.GetAccessConfigWindowsLogInfoCreateEventLevelEnum().ERROR,
 model.GetAccessConfigWindowsLogInfoCreateEventLevelEnum().CRITICAL,
 model.GetAccessConfigWindowsLogInfoCreateEventLevelEnum().VERBOSE,
 }
 timeOffsetWindowsLogInfo := &model.AccessConfigTimeOffset{
 Offset: int64(111),
 Unit: model.GetAccessConfigTimeOffsetUnitEnum().HOURL,
 }
 var listCategorysWindowsLogInfo = []model.AccessConfigWindowsLogInfoCreateCategorys{
```

```
model.GetAccessConfigWindowsLogInfoCreateCategoriesEnum().SYSTEM,
model.GetAccessConfigWindowsLogInfoCreateCategoriesEnum().SECURITY,
model.GetAccessConfigWindowsLogInfoCreateCategoriesEnum().SETUP,
}
windowsLogInfoAccessConfigDetail := &model.AccessConfigWindowsLogInfoCreate{
 Categories: listCategoriesWindowsLogInfo,
 TimeOffset: timeOffsetWindowsLogInfo,
 EventLevel: listEventLevelWindowsLogInfo,
}
modeMulti:= model.GetAccessConfigFormatMutilCreateModeEnum().TIME
valueMulti:= "YYYY-MM-DD hh:mm:ss"
multiFormat := &model.AccessConfigFormatMutilCreate{
 Mode: &modeMulti,
 Value: &valueMulti,
}
formatAccessConfigDetail := &model.AccessConfigFormatCreate{
 Multi: multiFormat,
}
var listBlackPathsAccessConfigDetail = []string{
 "/testxxx",
 "/tesxxx",
}
var listPathsAccessConfigDetail = []string{
 "/test/xxx",
 "/texxx",
}
accessConfigDetailbody := &model.AccessConfigDeatilCreate{
 Paths: &listPathsAccessConfigDetail,
 BlackPaths: &listBlackPathsAccessConfigDetail,
 Format: formatAccessConfigDetail,
 WindowsLogInfo: windowsLogInfoAccessConfigDetail,
}
request.Body = &model.CreateAccessConfigRequestBody{
 AccessConfigTag: &listAccessConfigTagbody,
 HostGroupInfo: hostGroupInfobody,
 LogInfo: logInfobody,
 AccessConfigDetail: accessConfigDetailbody,
 AccessConfigType: model.GetCreateAccessConfigRequestBodyAccessConfigTypeEnum().AGENT,
 AccessConfigName: "Tesxxx",
}
response, err := client.CreateAccessConfig(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                         |
|-----|----------------------------|
| 200 | 创建日志接入请求响应成功               |
| 400 | 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。     |

## 错误码

请参见[错误码](#)。

## 6.5.4 删除日志接入

### 功能介绍

删除日志接入

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v3/{project\_id}/lts/access-config

表 6-255 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取账号租户ID、项目资源集ID、日志组ID、日志流ID</a> 。<br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-256 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-257 请求 Body 参数

| 参数                    | 是否必选 | 参数类型             | 描述                             |
|-----------------------|------|------------------|--------------------------------|
| access_config_id_list | 是    | Array of strings | 日志接入ID列表<br>最小长度：36<br>最大长度：36 |

## 响应参数

状态码：200

表 6-258 响应 Body 参数

| 参数     | 参数类型                                              | 描述                          |
|--------|---------------------------------------------------|-----------------------------|
| result | Array of <a href="#">AccessConfigInfo</a> objects | 日志接入列表                      |
| total  | Long                                              | 日志接入总数<br>最小值：0<br>最大值：1000 |

表 6-259 AccessConfigInfo

| 参数                   | 参数类型                                            | 描述                                      |
|----------------------|-------------------------------------------------|-----------------------------------------|
| access_config_id     | String                                          | 日志接入ID<br>最小长度：36<br>最大长度：36            |
| access_config_name   | String                                          | 日志接入名称<br>最小长度：1<br>最大长度：64             |
| access_config_type   | String                                          | 日志接入类型。AGENT：主机接入枚举值：<br>• <b>AGENT</b> |
| create_time          | Long                                            | 创建时间<br>最小值：1<br>最大值：999999999999       |
| access_config_detail | <a href="#">AccessConfigDetailCreate</a> object | 日志接入详细信息                                |

| 参数                  | 参数类型                                               | 描述                   |
|---------------------|----------------------------------------------------|----------------------|
| log_info            | <a href="#">AccessConfigQueryLogInfo</a> object    | 日志接入日志详情             |
| host_group_info     | <a href="#">AccessConfigHostGroupIdList</a> object | 日志接入主机组ID列表          |
| access_config_tag   | Array of <a href="#">accessConfigTag</a> objects   | 标签信息。KEY不能重复,最多20个标签 |
| log_split           | Boolean                                            | 日志拆分                 |
| binary_collect      | Boolean                                            | 二进制收集                |
| cluster_id          | String                                             | CCE集群ID              |
| encoding_format     | String                                             | 编码格式, 默认UTF-8        |
| incremental_collect | Boolean                                            | 采集策略: 增量/全量          |
| processor_type      | String                                             | IC结构化解析类型            |
| demo_log            | String                                             | 示例日志                 |
| demo_fields         | Array of <a href="#">DemoFieldAccess</a> objects   | 示例日志解析字段             |
| processors          | Array of <a href="#">Processor</a> objects         | IC结构化解析器             |
| application_id      | String                                             | ServiceStage应用ID     |
| environment_id      | String                                             | ServiceStage环境ID     |
| component_id        | Array of strings                                   | ServiceStage组件ID     |

表 6-260 AccessConfigDeatilCreate

| 参数               | 参数类型                                                    | 描述                                                                                                                                                                        |
|------------------|---------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| paths            | Array of strings                                        | 采集路径。<br>1. 路径必须以/或者字母\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次`<br>CCE类型中 容器路径和主机路径必填, 标准输出不用<br>最小长度: 1<br>最大长度: 128<br>数组长度: 1 - 9 |
| black_paths      | Array of strings                                        | 采集路径黑名单。<br>1. 路径必须以/或者字母\开头<br>2. 不能包含特殊字符<>' " 且不能只输入/<br>3. 第一级目录不支持通配符*: 不能以/** /*开头<br>4.**只能出现一次<br>最小长度: 1<br>最大长度: 128<br>数组长度: 0 - 9                             |
| format           | <a href="#">AccessConfigFormatCreate</a> object         | 日志格式。single与multi必须选择一种。                                                                                                                                                  |
| windows_log_info | <a href="#">AccessConfigWindowsLogInfoCreate</a> object | 日志接入采集Windows事件日志                                                                                                                                                         |
| stdout           | Boolean                                                 | 标准输出开关, 仅CCE接入类型时使用                                                                                                                                                       |
| stderr           | Boolean                                                 | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                                                                 |
| pathType         | String                                                  | CCE接入类型, 仅CCE接入类型时使用<br>枚举值:<br><ul style="list-style-type: none"><li>• <b>HOST_FILE</b></li><li>• <b>CONTAINER_STDOUT</b></li><li>• <b>CONTAINER_FILE</b></li></ul>      |
| namespaceRegex   | String                                                  | K8s Namespace正则匹配, 仅CCE接入类型时使用                                                                                                                                            |

| 参数                 | 参数类型               | 描述                                               |
|--------------------|--------------------|--------------------------------------------------|
| podNameRegex       | String             | K8s Pod正则匹配，仅CCE接入类型时使用                          |
| containerNameRegex | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                         |
| includeLabels      | Map<String,String> | 容器 Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| excludeLabels      | Map<String,String> | 容器 Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| includeEnvs        | Map<String,String> | 环境变量白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| excludeEnvs        | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| logLabels          | Map<String,String> | 容器 Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logEnvs            | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用      |
| includeK8sLabels   | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| excludeK8sLabels   | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logK8s             | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用 |
| repeat_collect     | Boolean            | 强制重复采集                                           |

| 参数                      | 参数类型               | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| system_fields           | Array of strings   | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置<br/>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |
| custom_key_value        | Map<String,String> | <p>自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20</p> <p>键的长度限制为128，允许的字符有a-zA-Z0-9_-</p> <p>值的长度限制为1024</p> <p>若修改时传入此字段，将覆盖原有配置</p>                                                                                                                                                                                                                                                                                                                     |
| includeLabelsLogical    | String             | <p>容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| excludeLabelsLogical    | String             | <p>容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| includeK8sLabelsLogical | String             | <p>K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                      |

| 参数                      | 参数类型   | 描述                                                                                       |
|-------------------------|--------|------------------------------------------------------------------------------------------|
| excludeK8sLabelsLogical | String | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| includeEnvsLogical      | String | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |
| excludeEnvsLogical      | String | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |

表 6-261 AccessConfigFormatCreate

| 参数     | 参数类型                                                  | 描述         |
|--------|-------------------------------------------------------|------------|
| single | <a href="#">AccessConfigFormatSingleCreate</a> object | 日志接入格式单行日志 |
| multi  | <a href="#">AccessConfigFormatMutilCreate</a> object  | 日志接入格式多行日志 |

表 6-262 AccessConfigFormatSingleCreate

| 参数   | 参数类型   | 描述                                                                                                                                 |
|------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| mode | String | 单行日志。system：系统时间，wildcard：时间通配符。<br>枚举值： <ul style="list-style-type: none"><li>• <b>system</b></li><li>• <b>wildcard</b></li></ul> |

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                             |
|-------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | String | 日志时间。当mode为"system"，则填入当前时间戳。当mode为"wildcard"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss<br>最小长度：1<br>最大长度：64 |

表 6-263 AccessConfigFormatMutilCreate

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                         |
|-------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | String | 单行日志。time：日志时间，regular：正则模式。<br>枚举值： <ul style="list-style-type: none"><li>time</li><li>regular</li></ul>                                                                                                                                  |
| value | String | 日志时间。当mode为"regular"，则输入正则表达式当mode为"time"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss<br>最小长度：1<br>最大长度：64 |

表 6-264 AccessConfigWindowsLogInfoCreate

| 参数          | 参数类型                                          | 描述                                                                                                                                                                                                                                                                                |
|-------------|-----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | Array of strings                              | 采集Windows事件日志类型。<br>Application: 应用系统, System: 系统, Security: 安全, Setup: 启动<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>Application</b></li> <li>• <b>System</b></li> <li>• <b>Security</b></li> <li>• <b>Setup</b></li> </ul>                                          |
| time_offset | <a href="#">AccessConfigTimeOffset</a> object | 日志接入偏移时间                                                                                                                                                                                                                                                                          |
| event_level | Array of strings                              | 事件等级。information: info, warning: 告警, error: 错误, critical: 关键, verbose: 冗长<br>数组长度: 1 - 5<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>information</b></li> <li>• <b>warning</b></li> <li>• <b>error</b></li> <li>• <b>critical</b></li> <li>• <b>verbose</b></li> </ul> |

表 6-265 AccessConfigTimeOffset

| 参数     | 参数类型 | 描述                                                                                                                                           |
|--------|------|----------------------------------------------------------------------------------------------------------------------------------------------|
| offset | Long | 偏移时间。<br>当"unit"选择"day"时, 范围为1~7天。<br>当"unit"选择"hour"时, 范围为1~168小时。<br>当"unit"选择"sec"时, 范围为1~604800秒。<br>最小值: <b>1</b><br>最大值: <b>604800</b> |

| 参数   | 参数类型   | 描述                                                                                                                           |
|------|--------|------------------------------------------------------------------------------------------------------------------------------|
| unit | String | 偏移时间单位。day：天，hour：小时，sec：秒<br>枚举值：<br><ul style="list-style-type: none"> <li>• day</li> <li>• hour</li> <li>• sec</li> </ul> |

表 6-266 AccessConfigQueryLogInfo

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_group_id    | String | 日志组ID<br>最小长度：36<br>最大长度：36 |
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_group_name  | String | 日志组名称<br>最小长度：1<br>最大长度：128 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：128 |

表 6-267 AccessConfigHostGroupIdList

| 参数                 | 参数类型             | 描述                            |
|--------------------|------------------|-------------------------------|
| host_group_id_list | Array of strings | 主机组ID列表<br>最小长度：36<br>最大长度：36 |

表 6-268 accessConfigTag

| 参数    | 参数类型   | 描述                            |
|-------|--------|-------------------------------|
| key   | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | String | 标签Value<br>最小长度：0<br>最大长度：255 |

表 6-269 DemoFieldAccess

| 参数          | 参数类型   | 描述                |
|-------------|--------|-------------------|
| field_name  | String | 字段名称需和keys中字段保持一致 |
| field_value | String | 字段值               |

表 6-270 Processor

| 参数     | 参数类型                 | 描述                                                                                                                                                                                              |
|--------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | String               | 解析器类型 processor_regex 正则解析<br>processor_split_string 分词符<br>processor_json json解析器类型<br>processor_gotime自定义时间类型<br>processor_filter_regex日志过滤<br>processor_drop删除字段类型<br>processor_rename修改字段类型 |
| detail | <b>Detail</b> object | 解析器配置                                                                                                                                                                                           |

表 6-271 Detail

| 参数               | 参数类型             | 描述                           |
|------------------|------------------|------------------------------|
| source_key       | String           | 自定义时间key字段名称                 |
| regex            | String           | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段 |
| keys             | Array of strings | 字段名称列表 与field_name字段保持一致     |
| multi_line_regex | String           | 首行正则表达式                      |
| keep_source      | Boolean          | 是否上传原始日志                     |

| 参数                         | 参数类型               | 描述                                                   |
|----------------------------|--------------------|------------------------------------------------------|
| keep_source_if_parse_error | Boolean            | 是否上传解析失败日志                                           |
| split_sep                  | String             | 分隔符 自定义字符最大长度10, 自定义字符串最大长度30                        |
| split_type                 | String             | 分隔符类型: char-自定义字符; special_char-不可见字符; string-自定义字符串 |
| fields                     | Map<String,String> | 添加的字段列表: <字段名称 : 字段内容>                               |
| drop_keys                  | Array of strings   | 删除的字段列表                                              |
| source_keys                | Array of strings   | 字段重命名源字段名称列表                                         |
| dest_keys                  | Array of strings   | 字段重命名替换的字段名称列表                                       |
| expand_depth               | Integer            | json解析深度, 默认为1层最大4层                                  |
| expand_connector           | String             | json解析字段名称连接符                                        |
| source_format              | String             | 自定义时间格式                                              |
| source_value               | String             | 自定义时间字段value                                         |
| set_time                   | Boolean            | 是否开启自定义时间的开关                                         |
| include                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复       |
| exclude                    | Map<String,String> | 日志过滤黑名单规则 key长度最大为256, value最大长度为128, key不可以重复       |

状态码: 400

表 6-272 响应 Body 参数

| 参数         | 参数类型   | 描述                            |
|------------|--------|-------------------------------|
| error_code | String | 错误码<br>最小长度: 8<br>最大长度: 8     |
| error_msg  | String | 错误描述<br>最小长度: 1<br>最大长度: 1000 |

**状态码：500**

**表 6-273 响应 Body 参数**

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8     |
| error_msg  | String | 错误描述<br>最小长度：1<br>最大长度：1000 |

**请求示例**

删除日志接入

```
DELETE https://{endpoint}/v3/{project_id}/lts/access-config
/v3/{project_id}/lts/access-config
{"access_config_id_list":["xxxx","xxxx"]}
```

**响应示例**

**状态码：200**

删除日志接入请求响应成功

```
{
 "result": [{
 "access_config_detail": {
 "black_paths": ["/wjy/hei/tesxxx", "/wjy/hei/tesxxx"],
 "format": {
 "single": {
 "mode": "wildcard",
 "value": "1111"
 }
 },
 },
 "paths": ["/wjy/tesxxx", "/wjy/texxx", "/wjyxxxxx"],
 "windows_log_info": {
 "categorys": ["System", "Application", "Security", "Setup"],
 "event_level": ["information", "warning", "error", "critical", "verbose"],
 "time_offset": {
 "offset": 168,
 "unit": "hour"
 }
 }
 }],
 "access_config_id": "aa58d29e-21a9-4761-ba16-8xxxxx",
 "access_config_name": "采集Wjyxxxxxt2",
 "access_config_tag": [{
 "key": "xxx",
 "value": "xxx"
 }, {
 "key": "xxx1",
 "value": "xxx1"
 }],
 "access_config_type": "AGENT",
 "create_time": 1635043645628,
```

```
"host_group_info" : {
 "host_group_id_list" : ["de4dbed4-a3bc-4877-a7ee-0xxxxxx6"]
},
"log_info" : {
 "log_group_id" : "9a7e2183-2d6d-4732-9a9b-e89xxxxx0",
 "log_group_name" : "lts-groupxxxxka",
 "log_stream_id" : "c4de0538-53e6-41fd-b951-a8xxxxx58d7",
 "log_stream_name" : "lts-topic-txxxx"
}
}],
"total" : 1
}
```

### 状态码：400

非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.1807",
 "error_msg" : "Invalid access config id"
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0010",
 "error_msg" : "The system encountered an internal error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 删除日志接入

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class DeleteAccessConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);
```

```
LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
DeleteAccessConfigRequest request = new DeleteAccessConfigRequest();
try {
 DeleteAccessConfigResponse response = client.deleteAccessConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 删除日志接入

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = DeleteAccessConfigRequest()
 response = client.delete_access_config(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 删除日志接入

```
package main
```

```
import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.DeleteAccessConfigRequest{}
 response, err := client.DeleteAccessConfig(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                         |
|-----|----------------------------|
| 200 | 删除日志接入请求响应成功               |
| 400 | 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。     |

## 错误码

请参见[错误码](#)。

## 6.5.5 修改日志接入

### 功能介绍

修改日志接入

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v3/{project\_id}/lts/access-config

表 6-274 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取账号租户ID、项目资源集ID、日志组ID、日志流ID</a> 。<br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-275 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-276 请求 Body 参数

| 参数                   | 是否必选 | 参数类型                                                 | 描述                                                                      |
|----------------------|------|------------------------------------------------------|-------------------------------------------------------------------------|
| access_config_id     | 是    | String                                               | 日志接入ID<br>最小长度：36<br>最大长度：36                                            |
| access_config_name   | 否    | String                                               | 日志接入名称。<br>满足正则表达式：^(?!)(?!)(?!.*?.\$)[\u4e00-\u9fa5a-zA-Z0-9-]{1,64}\$ |
| access_config_detail | 否    | <a href="#">AccessConfigDetailUpdate</a> object      | 日志接入详细信息                                                                |
| host_group_info      | 否    | <a href="#">AccessConfigHostGroupInfoList</a> object | 日志接入主机组ID列表                                                             |
| access_config_tag    | 否    | Array of <a href="#">accessConfigTag</a> objects     | 标签信息。KEY不能重复,最多20个标签                                                    |
| log_split            | 否    | Boolean                                              | 日志拆分                                                                    |
| binary_collect       | 否    | Boolean                                              | 二进制采集                                                                   |
| cluster_id           | 否    | String                                               | CCE集群ID, CCE类型时, 为必填                                                    |
| incremental_collect  | 否    | Boolean                                              | 是否增量采集 true为是 false为否 ( 全量采集 )                                          |
| encoding_format      | 否    | String                                               | 编码格式, 默认UTF-8                                                           |
| processor_type       | 否    | String                                               | IC结构化解析类型                                                               |
| demo_log             | 否    | String                                               | 示例日志                                                                    |
| demo_fields          | 否    | Array of <a href="#">DemoFieldAccess</a> objects     | 示例日志解析字段                                                                |
| processors           | 否    | Array of <a href="#">Processor</a> objects           | IC结构化解析器                                                                |
| application_id       | 否    | String                                               | ServiceStage应用ID                                                        |
| environment_id       | 否    | String                                               | ServiceStage环境ID                                                        |

| 参数           | 是否必选 | 参数类型             | 描述               |
|--------------|------|------------------|------------------|
| component_id | 否    | Array of strings | ServiceStage组件ID |

表 6-277 AccessConfigDeatilUpdate

| 参数               | 是否必选 | 参数类型                                     | 描述                                                                                                                           |
|------------------|------|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| paths            | 否    | Array of strings                         | 1.路径必须以/或者字母:\开头<br>2.不能包含特殊字符<>' " 且不能只输入/<br>3.第一级目录不支持通配符*:不能以/** /*开头<br>4.**只能出现一次<br>5.最大数量为10<br>最小长度: 1<br>最大长度: 128 |
| black_paths      | 否    | Array of strings                         | 1.路径必须以/或者字母:\开头<br>2.不能包含特殊字符<>' " 且不能只输入/<br>3.第一级目录不支持通配符*:不能以/** /*开头<br>4.**只能出现一次<br>5.最大数量为10<br>最小长度: 1<br>最大长度: 128 |
| format           | 否    | AccessConfig FormatUpdate object         | 日志格式。若修改format, 则single与multi必须选择一种。                                                                                         |
| windows_log_info | 否    | AccessConfig WindowsLogInfoUpdate object | 日志接入采集Windows事件日志。当需要取消Windows日志时, 只需传入一个空的windows_log_info字段。                                                               |
| stdout           | 否    | Boolean                                  | 标准输出开关, 仅CCE接入类型时使用                                                                                                          |
| stderr           | 否    | Boolean                                  | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                    |

| 参数                 | 是否必选 | 参数类型               | 描述                                                                                                                                                               |
|--------------------|------|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| pathType           | 否    | String             | CCE接入类型，仅CCE接入类型时使用<br>枚举值： <ul style="list-style-type: none"><li>• <b>HOST_FILE</b></li><li>• <b>CONTAINER_STDOUT</b></li><li>• <b>CONTAINER_FILE</b></li></ul> |
| namespaceRegex     | 否    | String             | K8s Namespace正则匹配，仅CCE接入类型时使用                                                                                                                                    |
| podNameRegex       | 否    | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                                                                                                                                         |
| containerNameRegex | 否    | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                                                                                                                                         |
| includeLabels      | 否    | Map<String,String> | 容器 Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                   |
| excludeLabels      | 否    | Map<String,String> | 容器 Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                   |
| includeEnvs        | 否    | Map<String,String> | 环境变量白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                       |
| excludeEnvs        | 否    | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                       |
| logLabels          | 否    | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                      |
| logEnvs            | 否    | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                      |
| includeK8sLabels   | 否    | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                  |
| excludeK8sLabels   | 否    | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                  |
| logK8s             | 否    | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用                                                                                                                 |

| 参数                   | 是否必选 | 参数类型               | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------|------|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| system_fields        | 否    | Array of strings   | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置</p> <p>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |
| custom_key_value     | 否    | Map<String,String> | <p>自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20</p> <p>键的长度限制为128，允许的字符有a-zA-Z0-9_-</p> <p>值的长度限制为1024</p> <p>若修改时传入此字段，将覆盖原有配置</p>                                                                                                                                                                                                                                                                                                              |
| includeLabelsLogical | 否    | String             | <p>容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                 |

| 参数                      | 是否必选 | 参数类型   | 描述                                                                                       |
|-------------------------|------|--------|------------------------------------------------------------------------------------------|
| excludeLabelsLogical    | 否    | String | 容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效   |
| includeK8sLabelsLogical | 否    | String | K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| excludeK8sLabelsLogical | 否    | String | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| includeEnvsLogical      | 否    | String | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |
| excludeEnvsLogical      | 否    | String | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |

表 6-278 AccessConfigFormatUpdate

| 参数     | 是否必选 | 参数类型                                            | 描述         |
|--------|------|-------------------------------------------------|------------|
| single | 否    | <a href="#">AccessConfigFormatSingle</a> object | 日志接入格式单行日志 |
| multi  | 否    | <a href="#">AccessConfigFormatMutil</a> object  | 日志接入格式多行日志 |

表 6-279 AccessConfigFormatSingle

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                               |
|-------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | 是    | String | 单行日志。system: 系统时间, wildcard: 时间通配符。<br>枚举值:<br><ul style="list-style-type: none"> <li>• system</li> <li>• wildcard</li> </ul>                                                                                                                                    |
| value | 是    | String | 日志时间。当mode为"system", 则填入当前时间戳。当mode为"wildcard", 则时间通配符: 用日志打印时间来标识一条日志数据, 通过时间通配符来匹配日志, 每条日志的行首显示日志的打印时间; 如果日志中的时间格式为:<br>2019-01-01 23:59:59, 时间通配符应该填写为: YYYY-MM-DD hh:mm:ss; 如果日志中的时间格式为: 19-1-1 23:59:59, 时间通配符应该填写为: YY-M-D hh:mm:ss<br>最小长度: 1<br>最大长度: 64 |

表 6-280 AccessConfigFormatMutil

| 参数   | 是否必选 | 参数类型   | 描述                                                                                                                     |
|------|------|--------|------------------------------------------------------------------------------------------------------------------------|
| mode | 是    | String | 单行日志。time: 日志时间, regular: 正则模式。<br>枚举值:<br><ul style="list-style-type: none"> <li>• time</li> <li>• regular</li> </ul> |

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                                         |
|-------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | 是    | String | <p>日志时间。当mode为"regular", 则输入正则表达式<br/>当mode为"time", 则时间通配符: 用日志打印时间来标识一条日志数据, 通过时间通配符来匹配日志, 每条日志的行首显示日志的打印时间; 如果日志中的时间格式为: 2019-01-01 23:59:59, 时间通配符应该填写为: YYYY-MM-DD hh:mm:ss; 如果日志中的时间格式为: 19-1-1 23:59:59, 时间通配符应该填写为: YY-M-D hh:mm:ss</p> <p>最小长度: 1<br/>最大长度: 64</p> |

表 6-281 AccessConfigWindowsLogInfoUpdate

| 参数          | 是否必选 | 参数类型                          | 描述                                                                                                                                                                                                                                      |
|-------------|------|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | 否    | Array of strings              | <p>采集Windows事件日志类型。Application: 应用系统, System: 系统, Security: 安全, Setup: 启动</p> <p>枚举值:</p> <ul style="list-style-type: none"> <li>● Application</li> <li>● System</li> <li>● Security</li> <li>● Setup</li> </ul>                        |
| time_offset | 否    | AccessConfigTimeOffset object | 日志接入偏移时间                                                                                                                                                                                                                                |
| event_level | 否    | Array of strings              | <p>事件等级。information: info, warning: 告警, error: 错误, critical: 关键, verbose: 冗长</p> <p>枚举值:</p> <ul style="list-style-type: none"> <li>● information</li> <li>● warning</li> <li>● error</li> <li>● critical</li> <li>● verbose</li> </ul> |

表 6-282 AccessConfigTimeOffset

| 参数     | 是否必选 | 参数类型   | 描述                                                                                                                        |
|--------|------|--------|---------------------------------------------------------------------------------------------------------------------------|
| offset | 是    | Long   | 偏移时间。<br>当"unit"选择"day"时，范围为1~7天。<br>当"unit"选择"hour"时，范围为1~168小时。<br>当"unit"选择"sec"时，范围为1~604800秒。<br>最小值：1<br>最大值：604800 |
| unit   | 是    | String | 偏移时间单位。day：天，hour：小时，sec：秒<br>枚举值： <ul style="list-style-type: none"><li>• day</li><li>• hour</li><li>• sec</li></ul>     |

表 6-283 AccessConfigHostGroupIdList

| 参数                 | 是否必选 | 参数类型             | 描述                            |
|--------------------|------|------------------|-------------------------------|
| host_group_id_list | 是    | Array of strings | 主机组ID列表<br>最小长度：36<br>最大长度：36 |

表 6-284 accessConfigTag

| 参数    | 是否必选 | 参数类型   | 描述                            |
|-------|------|--------|-------------------------------|
| key   | 否    | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | 否    | String | 标签Value<br>最小长度：0<br>最大长度：255 |

表 6-285 DemoFieldAccess

| 参数          | 是否必选 | 参数类型   | 描述                |
|-------------|------|--------|-------------------|
| field_name  | 否    | String | 字段名称需和keys中字段保持一致 |
| field_value | 否    | String | 字段值               |

表 6-286 Processor

| 参数     | 是否必选 | 参数类型                 | 描述                                                                                                                                                                            |
|--------|------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | 否    | String               | 解析器类型 processor_regex 正则解析 processor_split_string 分词符 processor_json json解析器类型 processor_gotime自定义时间类型 processor_filter_regex日志过滤 processor_drop删除字段类型 processor_rename修改字段类型 |
| detail | 否    | <b>Detail</b> object | 解析器配置                                                                                                                                                                         |

表 6-287 Detail

| 参数                         | 是否必选 | 参数类型             | 描述                                                |
|----------------------------|------|------------------|---------------------------------------------------|
| source_key                 | 否    | String           | 自定义时间key字段名称                                      |
| regex                      | 否    | String           | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段                      |
| keys                       | 否    | Array of strings | 字段名称列表 与field_name字段保持一致                          |
| multi_line_regex           | 否    | String           | 首行正则表达式                                           |
| keep_source                | 否    | Boolean          | 是否上传原始日志                                          |
| keep_source_if_parse_error | 否    | Boolean          | 是否上传解析失败日志                                        |
| split_sep                  | 否    | String           | 分隔符 自定义字符最大长度10，自定义字符串最大长度30                      |
| split_type                 | 否    | String           | 分隔符类型：char-自定义字符；special_char-不可见字符；string-自定义字符串 |

| 参数               | 是否必选 | 参数类型               | 描述                                             |
|------------------|------|--------------------|------------------------------------------------|
| fields           | 否    | Map<String,String> | 添加的字段列表: <字段名称 : 字段内容>                         |
| drop_keys        | 否    | Array of strings   | 删除的字段列表                                        |
| source_keys      | 否    | Array of strings   | 字段重命名源字段名称列表                                   |
| dest_keys        | 否    | Array of strings   | 字段重命名替换的字段名称列表                                 |
| expand_depth     | 否    | Integer            | json解析深度, 默认为1层最大4层                            |
| expand_connector | 否    | String             | json解析字段名称连接符                                  |
| source_format    | 否    | String             | 自定义时间时间格式                                      |
| source_value     | 否    | String             | 自定义时间字段value                                   |
| set_time         | 否    | Boolean            | 是否开启自定义时间的开关                                   |
| include          | 否    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复 |
| exclude          | 否    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复 |

## 响应参数

状态码: 200

表 6-288 响应 Body 参数

| 参数                 | 参数类型   | 描述                             |
|--------------------|--------|--------------------------------|
| access_config_id   | String | 日志接入ID<br>最小长度: 36<br>最大长度: 36 |
| access_config_name | String | 日志接入名称<br>最小长度: 1<br>最大长度: 64  |

| 参数                   | 参数类型                                               | 描述                                          |
|----------------------|----------------------------------------------------|---------------------------------------------|
| access_config_type   | String                                             | 日志接入类型。AGENT：主机接入<br>枚举值：<br>• <b>AGENT</b> |
| create_time          | Long                                               | 创建时间<br>最小值：1<br>最大值：9999999999999          |
| access_config_detail | <a href="#">AccessConfigDetailCreate</a> object    | 日志接入详细信息                                    |
| log_info             | <a href="#">AccessConfigQueryLogInfo</a> object    | 日志接入日志详情                                    |
| host_group_info      | <a href="#">AccessConfigHostGroupIdList</a> object | 日志接入主机组ID列表                                 |
| access_config_tag    | Array of <a href="#">accessConfigTag</a> objects   | 标签信息。KEY不能重复,最多20个标签                        |
| log_split            | Boolean                                            | 日志拆分                                        |
| binary_collect       | Boolean                                            | 二进制收集                                       |
| cluster_id           | String                                             | CCE集群ID                                     |
| encoding_format      | String                                             | 编码格式，默认UTF-8                                |
| incremental_collect  | Boolean                                            | 采集策略：增量/全量                                  |
| processor_type       | String                                             | IC结构化解析类型                                   |
| demo_log             | String                                             | 示例日志                                        |
| demo_fields          | Array of <a href="#">DemoFieldAccess</a> objects   | 示例日志解析字段                                    |
| processors           | Array of <a href="#">Processor</a> objects         | IC结构化解析器                                    |
| application_id       | String                                             | ServiceStage应用ID                            |
| environment_id       | String                                             | ServiceStage环境ID                            |
| component_id         | Array of strings                                   | ServiceStage组件ID                            |

表 6-289 AccessConfigDeatilCreate

| 参数               | 参数类型                                                    | 描述                                                                                                                                                                                                                                                                 |
|------------------|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| paths            | Array of strings                                        | <p>采集路径。</p> <ol style="list-style-type: none"> <li>1. 路径必须以/或者字母:\开头</li> <li>2. 不能包含特殊字符&lt;&gt;' " 且不能只输入/</li> <li>3. 第一级目录不支持通配符*: 不能以/** /*开头</li> <li>4.**只能出现一次`</li> </ol> <p>CCE类型中 容器路径和主机路径必填, 标准输出不用</p> <p>最小长度: 1<br/>最大长度: 128<br/>数组长度: 1 - 9</p> |
| black_paths      | Array of strings                                        | <p>采集路径黑名单。</p> <ol style="list-style-type: none"> <li>1. 路径必须以/或者字母:\开头</li> <li>2. 不能包含特殊字符&lt;&gt;' " 且不能只输入/</li> <li>3. 第一级目录不支持通配符*: 不能以/** /*开头</li> <li>4.**只能出现一次</li> </ol> <p>最小长度: 1<br/>最大长度: 128<br/>数组长度: 0 - 9</p>                                 |
| format           | <a href="#">AccessConfigFormatCreate</a> object         | 日志格式。single与multi必须选择一种。                                                                                                                                                                                                                                           |
| windows_log_info | <a href="#">AccessConfigWindowsLogInfoCreate</a> object | 日志接入采集Windows事件日志                                                                                                                                                                                                                                                  |
| stdout           | Boolean                                                 | 标准输出开关, 仅CCE接入类型时使用                                                                                                                                                                                                                                                |
| stderr           | Boolean                                                 | 标准输出开关标准错误开关, 仅CCE接入类型时使用                                                                                                                                                                                                                                          |
| pathType         | String                                                  | <p>CCE接入类型, 仅CCE接入类型时使用枚举值:</p> <ul style="list-style-type: none"> <li>• <b>HOST_FILE</b></li> <li>• <b>CONTAINER_STDOUT</b></li> <li>• <b>CONTAINER_FILE</b></li> </ul>                                                                                           |
| namespaceRegex   | String                                                  | K8s Namespace正则匹配, 仅CCE接入类型时使用                                                                                                                                                                                                                                     |

| 参数                 | 参数类型               | 描述                                               |
|--------------------|--------------------|--------------------------------------------------|
| podNameRegex       | String             | K8s Pod正则匹配，仅CCE接入类型时使用                          |
| containerNameRegex | String             | K8s 容器名称正则匹配，仅CCE接入类型时使用                         |
| includeLabels      | Map<String,String> | 容器 Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| excludeLabels      | Map<String,String> | 容器 Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用   |
| includeEnvs        | Map<String,String> | 环境变量白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| excludeEnvs        | Map<String,String> | 环境变量黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用       |
| logLabels          | Map<String,String> | 容器 Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logEnvs            | Map<String,String> | 环境变量日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用      |
| includeK8sLabels   | Map<String,String> | K8s Label白名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| excludeK8sLabels   | Map<String,String> | K8s Label黑名单，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用  |
| logK8s             | Map<String,String> | K8s Label日志标签，最多支持创建30个，keyname不支持重名，仅CCE接入类型时使用 |
| repeat_collect     | Boolean            | 强制重复采集                                           |

| 参数                      | 参数类型               | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| system_fields           | Array of strings   | <p>系统内置字段：配置日志接入规则时，可以配置系统内置字段，上报日志后，每条日志数据的标签数据中将会有系统字段</p> <p>采集场景为主机文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、category、collectTime、__host_group__</p> <p>采集场景为K8S集群容器文件的内置字段为：<br/>hostName、hostId、hostIP、pathFile、hostIPv6、clusterId、podName、appName、containerName、nameSpace、category、collectTime、__host_group__、serviceID、podIp、clusterName、workloadType、__image_name__</p> <p>若修改时传入此字段，将覆盖原有配置<br/>首次配置元数据时，此字段必填，值为["pathFile"] (其中pathFile为默认的必传字段，其余字段可选)</p> |
| custom_key_value        | Map<String,String> | <p>自定义键值对：配置日志接入规则时，可以配置自定义键值对规则，上报日志后，每条日志数据的标签数据中将会有用户自定义的键值对字段，键值对数量不超过20</p> <p>键的长度限制为128，允许的字符有a-zA-Z0-9_-</p> <p>值的长度限制为1024</p> <p>若修改时传入此字段，将覆盖原有配置</p>                                                                                                                                                                                                                                                                                                                     |
| includeLabelsLogical    | String             | <p>容器 Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| excludeLabelsLogical    | String             | <p>容器 Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个容器 Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| includeK8sLabelsLogical | String             | <p>K8S Label白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效</p>                                                                                                                                                                                                                                                                                                                                                                                      |

| 参数                      | 参数类型   | 描述                                                                                       |
|-------------------------|--------|------------------------------------------------------------------------------------------|
| excludeK8sLabelsLogical | String | K8S Label黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个K8S Label黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效 |
| includeEnvsLogical      | String | 环境变量白名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量白名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |
| excludeEnvsLogical      | String | 环境变量黑名单生效逻辑，可选为AND，OR，不配置时默认为OR；当存在多个环境变量黑名单时的处理逻辑，AND表示同时满足才会生效，OR表示有一项满足就会生效           |

表 6-290 AccessConfigFormatCreate

| 参数     | 参数类型                                                  | 描述         |
|--------|-------------------------------------------------------|------------|
| single | <a href="#">AccessConfigFormatSingleCreate</a> object | 日志接入格式单行日志 |
| multi  | <a href="#">AccessConfigFormatMutilCreate</a> object  | 日志接入格式多行日志 |

表 6-291 AccessConfigFormatSingleCreate

| 参数   | 参数类型   | 描述                                                                                                                                    |
|------|--------|---------------------------------------------------------------------------------------------------------------------------------------|
| mode | String | 单行日志。system：系统时间，wildcard：时间通配符。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>system</b></li> <li>• <b>wildcard</b></li> </ul> |

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                                         |
|-------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | String | <p>日志时间。当mode为"system"，则填入当前时间戳。当mode为"wildcard"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss</p> <p>最小长度：1<br/>最大长度：64</p> |

表 6-292 AccessConfigFormatMutilCreate

| 参数    | 参数类型   | 描述                                                                                                                                                                                                                                                     |
|-------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode  | String | <p>单行日志。time：日志时间，regular：正则模式。</p> <p>枚举值：</p> <ul style="list-style-type: none"> <li>time</li> <li>regular</li> </ul>                                                                                                                                |
| value | String | <p>日志时间。当mode为"regular"，则输入正则表达式当mode为"time"，则时间通配符：用日志打印时间来标识一条日志数据，通过时间通配符来匹配日志，每条日志的行首显示日志的打印时间；如果日志中的时间格式为：2019-01-01 23:59:59，时间通配符应该填写为：YYYY-MM-DD hh:mm:ss；如果日志中的时间格式为：19-1-1 23:59:59，时间通配符应该填写为：YY-M-D hh:mm:ss</p> <p>最小长度：1<br/>最大长度：64</p> |

表 6-293 AccessConfigWindowsLogInfoCreate

| 参数          | 参数类型                          | 描述                                                                                                                                                                                                                                             |
|-------------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| categorys   | Array of strings              | 采集Windows事件日志类型。<br>Application: 应用系统, System: 系统, Security: 安全, Setup: 启动<br>枚举值:<br><ul style="list-style-type: none"> <li>• Application</li> <li>• System</li> <li>• Security</li> <li>• Setup</li> </ul>                                   |
| time_offset | AccessConfigTimeOffset object | 日志接入偏移时间                                                                                                                                                                                                                                       |
| event_level | Array of strings              | 事件等级。information: info, warning: 告警, error: 错误, critical: 关键, verbose: 冗长<br>数组长度: 1 - 5<br>枚举值:<br><ul style="list-style-type: none"> <li>• information</li> <li>• warning</li> <li>• error</li> <li>• critical</li> <li>• verbose</li> </ul> |

表 6-294 AccessConfigTimeOffset

| 参数     | 参数类型 | 描述                                                                                                                             |
|--------|------|--------------------------------------------------------------------------------------------------------------------------------|
| offset | Long | 偏移时间。<br>当"unit"选择"day"时, 范围为1~7天。<br>当"unit"选择"hour"时, 范围为1~168小时。<br>当"unit"选择"sec"时, 范围为1~604800秒。<br>最小值: 1<br>最大值: 604800 |

| 参数   | 参数类型   | 描述                                                                                                                           |
|------|--------|------------------------------------------------------------------------------------------------------------------------------|
| unit | String | 偏移时间单位。day：天，hour：小时，sec：秒<br>枚举值：<br><ul style="list-style-type: none"> <li>• day</li> <li>• hour</li> <li>• sec</li> </ul> |

表 6-295 AccessConfigQueryLogInfo

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_group_id    | String | 日志组ID<br>最小长度：36<br>最大长度：36 |
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_group_name  | String | 日志组名称<br>最小长度：1<br>最大长度：128 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：128 |

表 6-296 AccessConfigHostGroupIdList

| 参数                 | 参数类型             | 描述                            |
|--------------------|------------------|-------------------------------|
| host_group_id_list | Array of strings | 主机组ID列表<br>最小长度：36<br>最大长度：36 |

表 6-297 accessConfigTag

| 参数    | 参数类型   | 描述                            |
|-------|--------|-------------------------------|
| key   | String | 标签Key<br>最小长度：1<br>最大长度：128   |
| value | String | 标签Value<br>最小长度：0<br>最大长度：255 |

表 6-298 DemoFieldAccess

| 参数          | 参数类型   | 描述                |
|-------------|--------|-------------------|
| field_name  | String | 字段名称需和keys中字段保持一致 |
| field_value | String | 字段值               |

表 6-299 Processor

| 参数     | 参数类型                 | 描述                                                                                                                                                                                              |
|--------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type   | String               | 解析器类型 processor_regex 正则解析<br>processor_split_string 分词符<br>processor_json json解析器类型<br>processor_gotime自定义时间类型<br>processor_filter_regex日志过滤<br>processor_drop删除字段类型<br>processor_rename修改字段类型 |
| detail | <b>Detail</b> object | 解析器配置                                                                                                                                                                                           |

表 6-300 Detail

| 参数               | 参数类型             | 描述                           |
|------------------|------------------|------------------------------|
| source_key       | String           | 自定义时间key字段名称                 |
| regex            | String           | 正则解析正则表达式 单行完全正则，多行完全正则需填此字段 |
| keys             | Array of strings | 字段名称列表 与field_name字段保持一致     |
| multi_line_regex | String           | 首行正则表达式                      |
| keep_source      | Boolean          | 是否上传原始日志                     |

| 参数                         | 参数类型               | 描述                                                   |
|----------------------------|--------------------|------------------------------------------------------|
| keep_source_if_parse_error | Boolean            | 是否上传解析失败日志                                           |
| split_sep                  | String             | 分隔符 自定义字符最大长度10, 自定义字符串最大长度30                        |
| split_type                 | String             | 分隔符类型: char-自定义字符; special_char-不可见字符; string-自定义字符串 |
| fields                     | Map<String,String> | 添加的字段列表: <字段名称 : 字段内容>                               |
| drop_keys                  | Array of strings   | 删除的字段列表                                              |
| source_keys                | Array of strings   | 字段重命名源字段名称列表                                         |
| dest_keys                  | Array of strings   | 字段重命名替换的字段名称列表                                       |
| expand_depth               | Integer            | json解析深度, 默认为1层最大4层                                  |
| expand_connector           | String             | json解析字段名称连接符                                        |
| source_format              | String             | 自定义时间格式                                              |
| source_value               | String             | 自定义时间字段value                                         |
| set_time                   | Boolean            | 是否开启自定义时间的开关                                         |
| include                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复       |
| exclude                    | Map<String,String> | 日志过滤白名单规则 key长度最大为256, value最大长度为128, key不可以重复       |

状态码: 400

表 6-301 响应 Body 参数

| 参数         | 参数类型   | 描述                            |
|------------|--------|-------------------------------|
| error_code | String | 错误码<br>最小长度: 8<br>最大长度: 8     |
| error_msg  | String | 错误描述<br>最小长度: 1<br>最大长度: 1000 |

**状态码：500**

**表 6-302 响应 Body 参数**

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8     |
| error_msg  | String | 错误描述<br>最小长度：1<br>最大长度：1000 |

## 请求示例

### 修改日志接入（ECS）

```
PUT https://{endpoint}/v3/{project_id}/lts/access-config
{
 "access_config_id": "ed90802a-8475-4702-955e-e3ee16a5dde9",
 "access_config_detail": {
 "paths": ["/test/222", "/test/111"],
 "black_paths": [],
 "format": {
 "multi": {
 "mode": "regular",
 "value": "aaaa"
 }
 },
 "windows_log_info": {
 "categorys": ["Application", "System"],
 "time_offset": {
 "offset": 7,
 "unit": "day"
 },
 "event_level": ["information", "warning", "error", "critical", "verbose"]
 }
 },
 "host_group_info": {
 "host_group_id_list": ["de4dbed4-a3bc-4877-a7ee-096a2a63e036"]
 },
 "access_config_tag": [{
 "key": "xxx",
 "value": "xxx"
 }, {
 "key": "xxx1",
 "value": "xxx1"
 }]
}
```

## 响应示例

**状态码：200**

修改日志接入请求响应成功

```
{
 "access_config_detail": {
```

```
"black_paths": ["/wjy/hei/tesxxx", "/wjy/hei/tesxxx"],
"format": {
 "single": {
 "mode": "wildcard",
 "value": "1111"
 }
},
"paths": ["/wjy/tesxxx"],
"windows_log_info": {
 "categorys": ["System", "Application", "Security", "Setup"],
 "event_level": ["information", "warning", "error", "critical", "verbose"],
 "time_offset": {
 "offset": 168,
 "unit": "hour"
 }
}
},
"access_config_id": "aa58d29e-21a9-4761-ba16-8cxxxxd",
"access_config_name": "采集Wjy_xxxx2",
"access_config_tag": [{
 "key": "xxx",
 "value": "xxx"
}, {
 "key": "xxx1",
 "value": "xxx1"
}],
"access_config_type": "AGENT",
"create_time": 163504332654,
"host_group_info": {
 "host_group_id_list": ["de4dbed4-a3bc-4877-a7ee-09xxxxxx"]
},
"log_info": {
 "log_group_id": "9a7e2183-2d6d-4732-9axxxx49e0",
 "log_group_name": "lts-groupxxxxa",
 "log_stream_id": "c4de0538-53e6-41fd-b951-xxxx8d7",
 "log_stream_name": "lts-topixxx"
}
}
```

**状态码：400**

非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.1807",
 "error_msg": "Invalid access config id"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0010",
 "error_msg": "The system encountered an internal error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

修改日志接入（ECS）

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateAccessConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 UpdateAccessConfigRequest request = new UpdateAccessConfigRequest();
 UpdateAccessConfigRequestBody body = new UpdateAccessConfigRequestBody();
 List<AccessConfigTag> listbodyAccessConfigTag = new ArrayList<>();
 listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("xxx")
 .withValue("xxx")
);
 listbodyAccessConfigTag.add(
 new AccessConfigTag()
 .withKey("xxx1")
 .withValue("xxx1")
);
 List<String> listHostGroupInfoHostGroupIdList = new ArrayList<>();
 listHostGroupInfoHostGroupIdList.add("de4dbed4-a3bc-4877-a7ee-096a2a63e036");
 AccessConfigHostGroupInfoList hostGroupInfobody = new AccessConfigHostGroupInfoList();
 hostGroupInfobody.withHostGroupIdList(listHostGroupInfoHostGroupIdList);
 List<AccessConfigWindowsLogInfoUpdate.EventLevelEnum> listWindowsLogInfoEventLevel = new
 ArrayList<>();

 listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoUpdate.EventLevelEnum.fromValue("inform
 ation"));

 listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoUpdate.EventLevelEnum.fromValue("warnin
 g"));

 listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoUpdate.EventLevelEnum.fromValue("error"
));

 listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoUpdate.EventLevelEnum.fromValue("critical
 "));

 listWindowsLogInfoEventLevel.add(AccessConfigWindowsLogInfoUpdate.EventLevelEnum.fromValue("verbos
 e"));
 AccessConfigTimeOffsetCreate timeOffsetWindowsLogInfo = new AccessConfigTimeOffsetCreate();
 timeOffsetWindowsLogInfo.withOffset(7L)
```

```
.withUnit(AccessConfigTimeOffsetCreate.UnitEnum.fromValue("day"));
List<AccessConfigWindowsLogInfoUpdate.CategoriesEnum> listWindowsLogInfoCategories = new
ArrayList<>();

listWindowsLogInfoCategories.add(AccessConfigWindowsLogInfoUpdate.CategoriesEnum.fromValue("Applicat
ion"));

listWindowsLogInfoCategories.add(AccessConfigWindowsLogInfoUpdate.CategoriesEnum.fromValue("System"
));
AccessConfigWindowsLogInfoUpdate windowsLogInfoAccessConfigDetail = new
AccessConfigWindowsLogInfoUpdate();
windowsLogInfoAccessConfigDetail.withCategories(listWindowsLogInfoCategories)
.withTimeOffset(timeOffsetWindowsLogInfo)
.withEventLevel(listWindowsLogInfoEventLevel);
AccessConfigFormatMutil multiFormat = new AccessConfigFormatMutil();
multiFormat.withMode(AccessConfigFormatMutil.ModeEnum.fromValue("regular"))
.withValue("aaaa");
AccessConfigFormatUpdate formatAccessConfigDetail = new AccessConfigFormatUpdate();
formatAccessConfigDetail.withMulti(multiFormat);
List<String> listAccessConfigDetailPaths = new ArrayList<>();
listAccessConfigDetailPaths.add("/test/222");
listAccessConfigDetailPaths.add("/test/111");
AccessConfigDeatilUpdate accessConfigDetailbody = new AccessConfigDeatilUpdate();
accessConfigDetailbody.withPaths(listAccessConfigDetailPaths)
.withFormat(formatAccessConfigDetail)
.withWindowsLogInfo(windowsLogInfoAccessConfigDetail);
body.withAccessConfigTag(listbodyAccessConfigTag);
body.withHostGroupInfo(hostGroupInfobody);
body.withAccessConfigDetail(accessConfigDetailbody);
body.withAccessConfigId("ed90802a-8475-4702-955e-e3ee16a5dde9");
request.withBody(body);
try {
 UpdateAccessConfigResponse response = client.updateAccessConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 修改日志接入 ( ECS )

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"
```

```
credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = UpdateAccessConfigRequest()
 listAccessConfigTagbody = [
 AccessConfigTag(
 key="xxx",
 value="xxx"
),
 AccessConfigTag(
 key="xxx1",
 value="xxx1"
)
]
 listHostGroupIdListHostGroupInfo = [
 "de4dbed4-a3bc-4877-a7ee-096a2a63e036"
]
 hostGroupInfobody = AccessConfigHostGroupIdList(
 host_group_id_list=listHostGroupIdListHostGroupInfo
)
 listEventLevelWindowsLogInfo = [
 "information",
 "warning",
 "error",
 "critical",
 "verbose"
]
 timeOffsetWindowsLogInfo = AccessConfigTimeOffsetCreate(
 offset=7,
 unit="day"
)
 listCategorysWindowsLogInfo = [
 "Application",
 "System"
]
 windowsLogInfoAccessConfigDetail = AccessConfigWindowsLogInfoUpdate(
 categorys=listCategorysWindowsLogInfo,
 time_offset=timeOffsetWindowsLogInfo,
 event_level=listEventLevelWindowsLogInfo
)
 multiFormat = AccessConfigFormatMutil(
 mode="regular",
 value="aaaa"
)
 formatAccessConfigDetail = AccessConfigFormatUpdate(
 multi=multiFormat
)
 listPathsAccessConfigDetail = [
 "/test/222",
 "/test/111"
]
 accessConfigDetailbody = AccessConfigDeatilUpdate(
 paths=listPathsAccessConfigDetail,
 format=formatAccessConfigDetail,
 windows_log_info=windowsLogInfoAccessConfigDetail
)
 request.body = UpdateAccessConfigRequestBody(
 access_config_tag=listAccessConfigTagbody,
 host_group_info=hostGroupInfobody,
 access_config_detail=accessConfigDetailbody,
 access_config_id="ed90802a-8475-4702-955e-e3ee16a5dde9"
)
 response = client.update_access_config(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 修改日志接入 ( ECS )

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateAccessConfigRequest{}
 keyAccessConfigTag := "xxx"
 valueAccessConfigTag := "xxx"
 keyAccessConfigTag1 := "xxx1"
 valueAccessConfigTag1 := "xxx1"
 var listAccessConfigTagbody = []model.AccessConfigTag{
 {
 Key: &keyAccessConfigTag,
 Value: &valueAccessConfigTag,
 },
 {
 Key: &keyAccessConfigTag1,
 Value: &valueAccessConfigTag1,
 },
 }
 var listHostGroupIdListHostGroupInfo = []string{
 "de4dbed4-a3bc-4877-a7ee-096a2a63e036",
 }
 hostGroupInfoBody := &model.AccessConfigHostGroupIdList{
 HostGroupIdList: listHostGroupIdListHostGroupInfo,
 }
 var listEventLevelWindowsLogInfo = []model.AccessConfigWindowsLogInfoUpdateEventLevel{
 model.GetAccessConfigWindowsLogInfoUpdateEventLevelEnum().INFORMATION,
 model.GetAccessConfigWindowsLogInfoUpdateEventLevelEnum().WARNING,
 model.GetAccessConfigWindowsLogInfoUpdateEventLevelEnum().ERROR,
 model.GetAccessConfigWindowsLogInfoUpdateEventLevelEnum().CRITICAL,
```

```
 model.GetAccessConfigWindowsLogInfoUpdateEventLevelEnum().VERBOSE,
 }
 timeOffsetWindowsLogInfo := &model.AccessConfigTimeOffsetCreate{
 Offset: int64(7),
 Unit: model.GetAccessConfigTimeOffsetCreateUnitEnum().DAY,
 }
 var listCategorysWindowsLogInfo = []model.AccessConfigWindowsLogInfoUpdateCategorys{
 model.GetAccessConfigWindowsLogInfoUpdateCategorysEnum().APPLICATION,
 model.GetAccessConfigWindowsLogInfoUpdateCategorysEnum().SYSTEM,
 }
 windowsLogInfoAccessConfigDetail := &model.AccessConfigWindowsLogInfoUpdate{
 Categorys: &listCategorysWindowsLogInfo,
 TimeOffset: timeOffsetWindowsLogInfo,
 EventLevel: &listEventLevelWindowsLogInfo,
 }
 multiFormat := &model.AccessConfigFormatMutil{
 Mode: model.GetAccessConfigFormatMutilModeEnum().REGULAR,
 Value: "aaaa",
 }
 formatAccessConfigDetail := &model.AccessConfigFormatUpdate{
 Multi: multiFormat,
 }
 var listPathsAccessConfigDetail = []string{
 "/test/222",
 "/test/111",
 }
 accessConfigDetailbody := &model.AccessConfigDeatilUpdate{
 Paths: &listPathsAccessConfigDetail,
 Format: formatAccessConfigDetail,
 WindowsLogInfo: windowsLogInfoAccessConfigDetail,
 }
 request.Body = &model.UpdateAccessConfigRequestBody{
 AccessConfigTag: &listAccessConfigTagbody,
 HostGroupInfo: hostGroupInfobody,
 AccessConfigDetail: accessConfigDetailbody,
 AccessConfigId: "ed90802a-8475-4702-955e-e3ee16a5dde9",
 }
 response, err := client.UpdateAccessConfig(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                         |
|-----|----------------------------|
| 200 | 修改日志接入请求响应成功               |
| 400 | 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。     |

## 错误码

请参见[错误码](#)。

## 6.6 日志转储

### 6.6.1 创建日志转储（旧版）

#### 功能介绍

该接口用于将指定的一个或多个日志流的日志转储到OBS服务。

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v2/{project\_id}/log-dump/obs

表 6-303 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

#### 请求参数

表 6-304 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-305 请求 Body 参数

| 参数              | 是否必选 | 参数类型             | 描述                                                                                                                                            |
|-----------------|------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| log_group_id    | 是    | String           | 日志组id。<br>最小长度：36<br>最大长度：36                                                                                                                  |
| log_stream_ids  | 是    | Array of strings | 日志流id列表, 可以指定一个或多个日志流进行obs周期性转储                                                                                                               |
| obs_bucket_name | 是    | String           | obs 桶名称。<br>最小长度：3<br>最大长度：63                                                                                                                 |
| type            | 是    | String           | 周期性转储, 必须填 cycle。<br>最小长度：5<br>最大长度：5                                                                                                         |
| storage_format  | 是    | String           | 转储格式 RAW/JSON, 默认为 RAW。<br>最小长度：3<br>最大长度：4                                                                                                   |
| switch_on       | 否    | Boolean          | 是否开启转储 true/false, 默认为 true                                                                                                                   |
| prefix_name     | 否    | String           | 转储至OBS桶中的日志文件前缀。<br>最小长度：0<br>最大长度：64                                                                                                         |
| dir_prefix_name | 否    | String           | 自定义文件夹路径。<br>最小长度：0<br>最大长度：64                                                                                                                |
| period          | 是    | Integer          | 转储周期的长度<br>枚举值： <ul style="list-style-type: none"><li>• 1</li><li>• 2</li><li>• 3</li><li>• 5</li><li>• 6</li><li>• 12</li><li>• 30</li></ul> |

| 参数          | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                         |
|-------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| period_unit | 是    | String | 转储周期的单位。> period与 period_unit拼接后必须在该列表中<br>["2min","5min","30min","1hour","3hour","6hour","12hour"]<br>最小长度：3<br>最大长度：4<br>枚举值：<br><ul style="list-style-type: none"> <li>"min"</li> <li>"hour"</li> </ul> |

## 响应参数

状态码：201

表 6-306 响应 Body 参数

| 参数              | 参数类型   | 描述                                      |
|-----------------|--------|-----------------------------------------|
| log_dump_obs_id | String | 转储id。<br>缺省值：None<br>最小长度：36<br>最大长度：36 |

状态码：400

表 6-307 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                            |
|------------|--------|---------------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0007</li> </ul>                                    |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>The request body format must be json</li> </ul> |

状态码：403

表 6-308 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

状态码：500

表 6-309 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

## 请求示例

创建日志转储

POST https://{endpoint}/v2/{project\_id}/log-dump/obs

```
{
 "log_group_id": "d9dba9f3-xxxx-48bd-xxxx-xxxxa24a8053",
 "log_stream_ids": ["45e7f609-xxxx-4cd3-835b-xxxx4a124718"],
 "obs_bucket_name": "lts-test",
 "type": "cycle",
 "storage_format": "RAW",
 "switch_on": "true",
 "prefix_name": "fileprefixname",
 "dir_prefix_name": "dirprefixname",
 "period": 5,
 "period_unit": "min"
}
```

## 响应示例

状态码：200

请求响应成功。

- 当前日志组不存在。

```
{
 "error_code": "LTS.0201",
 "error_msg": "The log group does not existed"
}
```

- 当前日志流不存在。

```
{
 "error_code": "LTS.0208",
 "error_msg": "Log stream id does not exist: 632b9bdc-5afd-4666-a5de-2579f8b80314-"
}
```

- 当前OBS桶不存在。

```
{
 "error_code": "LTS.0416",
 "error_msg": "obs bucket does not exist: zhuanchu"
}
```

- 转储时日志流ID已经被关联。

```
{
 "error_code": "LTS.0207",
 "error_msg": "Log stream id is associated by transfer: 632b9bdc-5afd-4666-a5de-2579f8b80314"
}
```

- 转储类型没有在列表中。

```
{
 "error_code": "LTS.1901",
 "error_msg": "type is not in the list [cycle]"
}
```

- 转储格式没有在列表中。

```
{
 "error_code": "LTS.1901",
 "error_msg": "storage_format is not in the list [RAW, JSON]"
}
```

- 转储周期没有在列表中。

```
{
 "error_code": "LTS.1901",
 "error_msg": "period+period_unit is not in the list [2min, 5min, 30min, 1hour, 3hour, 6hour, 12hour]"
}
```

- 转储单位没有在列表中。

```
{
 "error_code": "LTS.1901",
 "error_msg": "period_unit is not in the list [min, hour]"
}
```

- 转储日志文件前缀无效，请确认是否按要求提供。

```
{
 "error_code": "LTS.1902",
 "error_msg": "prefix_name is invalid, please verify if it's provided as required"
}
```

- 自定义转储路径名前缀无效，请确认是否按要求提供。

```
{
 "error_code": "LTS.1902",
 "error_msg": "dir_prefix_name is invalid, please verify if it's provided as required"
}
```

### 状态码：201

请求响应成功。

```
{
 "log_dump_obs_id": "45fdc36b-xxxx-4567-xxxx-559xxxxdf968"
}
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。
{
 "error_code": "LTS.0007",
 "error_msg": "The request body format must be json"
}
```

#### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

#### 状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
InternalServerError。表明服务端能被请求访问到，但是服务内部出错。
{
 "error_code": "LTS.0010",
 "error_msg": "Internal Server Error"}
}
```

## 状态码

| 状态码 | 描述                                                                                           |
|-----|----------------------------------------------------------------------------------------------|
| 200 | 请求响应成功。                                                                                      |
| 201 | 请求响应成功。                                                                                      |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。                                                |
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到，但是服务内部出错。                                               |

## 错误码

请参见[错误码](#)。

## 6.6.2 创建日志转储（新版）

### 功能介绍

该接口用于创建OBS转储，DIS转储，DMS转储。

## 调用方法

请参见[如何调用API](#)。

## URI

POST /v2/{project\_id}/transfers

表 6-310 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-311 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-312 请求 Body 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                |
|--------------|------|--------|---------------------------------------------------|
| log_group_id | 是    | String | 日志组ID<br>最小长度：36<br>最大长度：36<br>最小长度：36<br>最大长度：36 |

| 参数                | 是否必选 | 参数类型                                        | 描述      |
|-------------------|------|---------------------------------------------|---------|
| log_streams       | 是    | Array of <a href="#">LogStreams</a> objects | 日志流ID集合 |
| log_transfer_info | 是    | <a href="#">log_transfer_info</a> object    | 日志转储信息  |

表 6-313 LogStreams

| 参数              | 是否必选 | 参数类型   | 描述                            |
|-----------------|------|--------|-------------------------------|
| log_stream_id   | 是    | String | 日志流ID<br>最小长度: 36<br>最大长度: 36 |
| log_stream_name | 否    | String | 日志流名称                         |

表 6-314 log\_transfer\_info

| 参数                 | 是否必选 | 参数类型   | 描述                                                                                                                                                                   |
|--------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_type  | 是    | String | 日志转储类型。OBS指OBS日志转储，DIS指DIS日志转储，DMS指DMS日志转储。                                                                                                                          |
| log_transfer_mode  | 是    | String | 日志转储方式。cycle是指周期性转储，realTime是指实时转储。OBS转储只支持"cycle"，DIS转储和DMS转储只支持"realTime"<br>枚举值：<br><ul style="list-style-type: none"> <li>• cycle</li> <li>• realTime</li> </ul> |
| log_storage_format | 是    | String | 日志转储格式。只支持"RAW"，"JSON"。RAW是指原始日志格式，JSON是指JSON日志格式。<br>枚举值：<br><ul style="list-style-type: none"> <li>• JSON</li> <li>• RAW</li> </ul>                                |

| 参数                  | 是否必选 | 参数类型                                       | 描述                                                                                                                                                                                                        |
|---------------------|------|--------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_status | 是    | String                                     | 日志转储状态，只支持 "ENABLE", "DISABLE", "EXCEPTION"。ENABLE是指日志转储开启状态，DISABLE是指日志转储关闭状态，EXCEPTION是指日志转储异常状态<br>枚举值： <ul style="list-style-type: none"><li>• ENABLE</li><li>• DISABLE</li><li>• EXCEPTION</li></ul> |
| log_agency_transfer | 否    | <a href="#">log_agency_transfer</a> object | 委托转储信息。若配置委托转储，则需要输入该参数                                                                                                                                                                                   |
| log_transfer_detail | 是    | <a href="#">log_transfer_detail</a> object | 日志转储详细信息                                                                                                                                                                                                  |

表 6-315 log\_agency\_transfer

| 参数                   | 是否必选 | 参数类型   | 描述                                            |
|----------------------|------|--------|-----------------------------------------------|
| agency_domain_id     | 是    | String | 委托方账号ID<br>最小长度：1<br>最大长度：128                 |
| agency_domain_name   | 是    | String | 委托方账号名称<br>最小长度：1<br>最大长度：128                 |
| agency_name          | 是    | String | 委托方配置的委托名称<br>最小长度：1<br>最大长度：128              |
| agency_project_id    | 是    | String | 委托方项目ID<br>最小长度：32<br>最大长度：32                 |
| be_agency_domain_id  | 是    | String | 被委托方账号ID，实际配置转储的账号ID<br>最小长度：1<br>最大长度：128    |
| be_agency_project_id | 是    | String | 被委托方项目ID，实际配置转储的账号的项目ID<br>最小长度：32<br>最大长度：32 |

表 6-316 log\_transfer\_detail

| 参数               | 是否必选 | 参数类型    | 描述                                                                                                                                                                                                                                                                                |
|------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period       | 是    | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• 1</li><li>• 2</li><li>• 3</li><li>• 5</li><li>• 6</li><li>• 12</li><li>• 30</li></ul> |
| obs_period_unit  | 是    | String  | OBS转储单位。当创建OBS转储时，必填此参数。与obs_period组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• min</li><li>• hour</li></ul>                                                               |
| obs_bucket_name  | 是    | String  | OBS转储日志桶名称。当创建OBS转储时，必填此参数。<br>最小长度：3<br>最大长度：63                                                                                                                                                                                                                                  |
| obs_encrypted_id | 否    | String  | OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志加密桶则必须填写该参数，若OBS转储日志桶则不需要此参数<br>最小长度：36<br>最大长度：36                                                                                                                                                                                          |

| 参数                   | 是否必选 | 参数类型    | 描述                                                                                          |
|----------------------|------|---------|---------------------------------------------------------------------------------------------|
| obs_dir_pre_fix_name | 否    | String  | OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。<br>最小长度：1<br>最大长度：64                                      |
| obs_prefix_name      | 否    | String  | OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。<br>最小长度：1<br>最大长度：64                                       |
| obs_time_zone        | 否    | String  | OBS转储时区。参数选择参考OBS转储时区表。如果选择该参数，则必须选择obs_time_zone_id。                                       |
| obs_time_zone_id     | 否    | String  | OBS转储时区ID。参数选择参考OBS转储时区表。如果选择该参数，则必须选择obs_time_zone。                                        |
| dis_id               | 否    | String  | DIS转储通道ID。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128                                            |
| dis_name             | 否    | String  | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：64                                             |
| kafka_id             | 否    | String  | DMS转储kafka ID。当创建DMS转储时，必填此参数。创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例    |
| kafka_topic          | 否    | String  | DMS转储kafka topic。当创建DMS转储时，必填此参数。创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例 |
| obs_eps_id           | 否    | String  | OBS企业项目ID。                                                                                  |
| obs_encrypted_enable | 否    | Boolean | OBS是否开启加密。                                                                                  |

| 参数                  | 是否必选 | 参数类型             | 描述                                                                                                                                                          |
|---------------------|------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tags                | 否    | Array of strings | 若开启tag投递, 该字段必须包含主机信息: hostIP、hostId、hostName、pathFile、collectTime;<br>公共字段有: logStreamName、regionName、logGroupName、projectId, 为可选填; 开启转储标签: streamTag, 可选填 |
| lts_tags            | 否    | Array of strings | dms转储JSON格式选填, 可以转储tag字段                                                                                                                                    |
| stream_tags         | 否    | Array of strings | dms转储JSON格式选填, 可以转储日志流标签字段                                                                                                                                  |
| struct_fields       | 否    | Array of strings | dms转储JSON格式选填, 可以转储结构化字段                                                                                                                                    |
| invalid_field_value | 否    | String           | dms转储JSON格式选填, 无效字段填充                                                                                                                                       |

## 响应参数

状态码: 200

表 6-317 响应 Body 参数

| 参数                | 参数类型                                              | 描述                             |
|-------------------|---------------------------------------------------|--------------------------------|
| log_group_id      | String                                            | 日志组ID<br>最小长度: 36<br>最大长度: 36  |
| log_group_name    | String                                            | 日志组名称<br>最小长度: 1<br>最大长度: 64   |
| log_streams       | Array of <a href="#">log_streams</a> objects      | 日志流集合                          |
| log_transfer_id   | String                                            | 日志转储ID<br>最小长度: 36<br>最大长度: 36 |
| log_transfer_info | <a href="#">log_transfer_info_RespBody</a> object | 日志转储信息                         |

表 6-318 log\_streams

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：64  |

表 6-319 log\_transfer\_info\_RespBody

| 参数                  | 参数类型                              | 描述                                                                                                                                                                  |
|---------------------|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_agency_transfer | <b>log_agency_transfer</b> object | 委托转储信息。若转储为委托转储，则会返回该参数                                                                                                                                             |
| log_create_time     | Integer                           | 日志转储创建时间<br>最小值：0<br>最大值：2147483647                                                                                                                                 |
| log_storage_format  | String                            | 日志转储格式。只支持"RAW", "JSON"。RAW是指原始日志格式，JSON是指JSON日志格式。OBS转储和DIS转储支持JSON和RAW，DMS转储仅支持RAW<br>枚举值：<br><ul style="list-style-type: none"> <li>JSON</li> <li>RAW</li> </ul> |
| log_transfer_detail | <b>TransferDetail</b> object      | 日志转储详细信息                                                                                                                                                            |
| log_transfer_mode   | String                            | 日志转储方式。cycle是指周期性转储，realTime是指实时转储。OBS转储只支持"cycle"，DIS转储和DMS转储只支持"realTime"。<br>枚举值：<br><ul style="list-style-type: none"> <li>cycle</li> <li>realTime</li> </ul>   |

| 参数                  | 参数类型   | 描述                                                                                                                                                                          |
|---------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_status | String | 日志转储状态，ENABLE是指日志转储开启状态，DISABLE是指日志转储关闭状态，EXCEPTION是指日志转储异常状态<br>枚举值：<br><ul style="list-style-type: none"> <li>• ENABLE</li> <li>• DISABLE</li> <li>• EXCEPTION</li> </ul> |
| log_transfer_type   | String | 日志转储类型。OBS指OBS日志转储，DIS指DIS日志转储，DMS指DMS日志转储。<br>枚举值：<br><ul style="list-style-type: none"> <li>• OBS</li> <li>• DIS</li> <li>• DMS</li> </ul>                                |

表 6-320 log\_agency\_transfer

| 参数                  | 参数类型   | 描述                                         |
|---------------------|--------|--------------------------------------------|
| agency_domain_id    | String | 委托方账号ID<br>最小长度：1<br>最大长度：128              |
| agency_domain_name  | String | 委托方账号名称<br>最小长度：1<br>最大长度：128              |
| agency_name         | String | 委托方配置的委托名称<br>最小长度：1<br>最大长度：128           |
| agency_project_id   | String | 委托方项目ID<br>最小长度：32<br>最大长度：32              |
| be_agency_domain_id | String | 被委托方账号ID，实际配置转储的账号ID<br>最小长度：1<br>最大长度：128 |

| 参数                   | 参数类型   | 描述                                            |
|----------------------|--------|-----------------------------------------------|
| be_agency_project_id | String | 被委托方项目ID，实际配置转储的账号的项目ID<br>最小长度：32<br>最大长度：32 |

表 6-321 TransferDetail

| 参数              | 参数类型    | 描述                                                                                                                                                                                                                                                                                |
|-----------------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period      | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• 1</li><li>• 2</li><li>• 3</li><li>• 5</li><li>• 6</li><li>• 12</li><li>• 30</li></ul> |
| obs_period_unit | String  | OBS转储单位。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• "min"</li><li>• "hour"</li></ul>                                                      |
| obs_bucket_name | String  | OBS日志桶名称。当创建OBS转储时，必填此参数。<br>最小长度：3<br>最大长度：63<br>最小长度：3<br>最大长度：63                                                                                                                                                                                                               |

| 参数                   | 参数类型   | 描述                                                                                                                                                 |
|----------------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_encrypted_id     | String | OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志桶加密则必须填写该参数，若OBS转储日志桶则不需要此参数。<br>最小长度：36<br>最大长度：36<br>最小长度： <b>36</b><br>最大长度： <b>36</b>                    |
| obs_dir_pre_fix_name | String | OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。<br>正则约束：<br>^(/)?([a-zA-Z0-9.-]+)(/[a-zA-Z0-9-]+)?/?\$<br>最小长度：1<br>最大长度：64<br>最小长度： <b>1</b><br>最大长度： <b>64</b> |
| obs_prefix_name      | String | OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。<br>正则约束：<br>^[a-zA-Z0-9._]*\$<br>最小长度：1<br>最大长度：64<br>最小长度： <b>1</b><br>最大长度： <b>64</b>                           |
| obs_time_zone        | String | OBS转储时区，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone_id。                                                                           |
| obs_time_zone_id     | String | OBS转储时区ID，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone。                                                                            |
| dis_id               | String | DIS转储通道ID。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度： <b>1</b><br>最大长度： <b>128</b>                                                             |

| 参数                   | 参数类型             | 描述                                                                                                                                                       |
|----------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| dis_name             | String           | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                                                                   |
| kafka_id             | String           | DMS转储kafka ID。当创建DMS转储时，必填此参数。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：36<br>最大长度：36                                       |
| kafka_topic          | String           | DMS转储kafka topic。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：1<br>最大长度：128                                                    |
| obs_transfer_path    | String           | OBS转储路径，指OBS日志桶中的路径。<br>最小长度：0<br>最大长度：256                                                                                                               |
| obs_eps_id           | String           | OBS企业项目ID。<br>最小长度：0<br>最大长度：128                                                                                                                         |
| obs_encrypted_enable | Boolean          | OBS是否开启加密。<br>枚举值： <ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>                                                                |
| tags                 | Array of strings | 若开启tag投递，该字段必须包含主机信息：hostIP、hostId、hostName、pathFile、collectTime；<br>公共字段有：logStreamName、regionName、logGroupName、projectId，为可选填；<br>开启转储标签：streamTag，可选填 |

状态码：400

表 6-322 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-323 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

- 创建OBS转储

POST https://{endpoint}/v2/{project\_id}/transfers

```
{
 "log_group_id": "8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxx",
 "log_streams": [{
 "log_stream_id": "c776e1a7-8548-430a-afe5-xxxxxxxxxxxx"
 }],
 "log_transfer_info": {
 "log_transfer_type": "OBS",
 "log_transfer_mode": "xxxxx",
 "log_storage_format": "XXX",
 "log_transfer_status": "XXXXX",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "obs_period": 2,
 "obs_period_unit": "min",
 "obs_bucket_name": "xxxxx",
 "obs_encrypted_id": "1bd90032-1424-481f-8558-ba49854xxxxx",
 "obs_dir_pre_fix_name": "/xx",
 "obs_prefix_name": "xxxxx",
 "obs_time_zone": "UTC+01:00",
 "obs_time_zone_id": "Africa/Lagos"
 }
 }
}
```

- 创建DIS转储

POST https://{endpoint}/v2/{project\_id}/transfers

```
{
 "log_group_id": "8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxx",
 "log_streams": [{
 "log_stream_id": "c776e1a7-8548-430a-afe5-xxxxxxxxxxxx"
 }],
 "log_transfer_info": {
 "log_transfer_type": "DIS",
 "log_transfer_mode": "xxxxx",
 "log_storage_format": "XXX",
 "log_transfer_status": "XXXXX",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "dis_id": "i1y8vfMTvf4LQzxxxxx",
 "dis_name": "xxxxx"
 }
 }
}
```

## 响应示例

**状态码：200**

创建转储请求响应成功。

- 当创建OBS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "obs_period": 2,
 "obs_prefix_name": "",
 "obs_period_unit": "min",
 "obs_transfer_path": "/0002/LogTanks/cn-north-7/",
 "obs_bucket_name": "0002",
 "obs_encrypted_enable": false,
 "obs_dir_pre_fix_name": "",
 "obs_time_zone": "UTC+01:00",
 "obs_time_zone_id": "Africa/Lagos",
 "tags": []
 }
 },
 "log_transfer_mode": "cycle",
}
```

```
"log_transfer_status": "ENABLE",
"log_transfer_type": "OBS"
}
```

- 当创建DIS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "dis_id": "xxxxx",
 "dis_name": "xxxxxx",
 "tags": []
 },
 "log_transfer_mode": "cycle",
 "log_transfer_status": "ENABLE",
 "log_transfer_type": "OBS"
 }
}
```

- 当创建DMS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "kafka_id": "xxxxxx",
 "kafka_topic": "xxxxx",
 "tags": []
 },
 "log_transfer_mode": "cycle",
 "log_transfer_status": "ENABLE",
 "log_transfer_type": "OBS"
 }
}
```

**状态码：400**

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.0207",
 "error_msg" : "The log stream is associated by transfer"
}
```

**状态码：500**

InternalServerError。表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0207",
 "error_msg" : "The log stream is associated by transfer"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 创建OBS转储

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateTransferSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 CreateTransferRequest request = new CreateTransferRequest();
 CreateTransferRequestBody body = new CreateTransferRequestBody();
 TransferDetail logTransferDetailLogTransferInfo = new TransferDetail();
 logTransferDetailLogTransferInfo.withObsPeriod(TransferDetail.ObsPeriodEnum.NUMBER_2)
 .withObsEncryptedId("1bd90032-1424-481f-8558-ba49854xxxxx")
 .withObsPrefixName("xxxxx")
 .withObsPeriodUnit("min")
 .withObsBucketName("xxxxx")
 .withObsDirPreFixName("/xx")
 .withObsTimeZone("UTC+01:00")
 }
}
```

```
.withObsTimeZoneId("Africa/Lagos");
CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer logAgencyTransferLogTransferInfo
= new CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer();
logAgencyTransferLogTransferInfo.withAgencyDomainId("1d26cc8c86a840e28a4f8d0d078xxxxx")
.withAgencyDomainName("paas_apm_z004xxxxx_xx")
.withAgencyName("test20210325")
.withAgencyProjectId("2a473356cca5487f8373be891bfxxxxx")
.withBeAgencyDomainId("1d26cc8c86a840e28a4f8d0d078xxxxx")
.withBeAgencyProjectId("2a473356cca5487f8373be891bfxxxxx");
CreateTransferRequestBodyLogTransferInfo logTransferInfobody = new
CreateTransferRequestBodyLogTransferInfo();
logTransferInfobody.withLogTransferType("OBS")
.withLogTransferMode(CreateTransferRequestBodyLogTransferInfo.LogTransferModeEnum.from
Value("xxxxx"))
.withLogStorageFormat(CreateTransferRequestBodyLogTransferInfo.LogStorageFormatEnum.fr
omValue("XXX"))
.withLogTransferStatus(CreateTransferRequestBodyLogTransferInfo.LogTransferStatusEnum.fr
omValue("XXXXX"))
.withLogAgencyTransfer(logAgencyTransferLogTransferInfo)
.withLogTransferDetail(logTransferDetailLogTransferInfo);
List<CreateTransferRequestBodyLogStreams> listbodyLogStreams = new ArrayList<>();
listbodyLogStreams.add(
 new CreateTransferRequestBodyLogStreams()
 .withLogStreamId("c776e1a7-8548-430a-afe5-xxxxxxxxxxxxx")
);
body.withLogTransferInfo(logTransferInfobody);
body.withLogStreams(listbodyLogStreams);
body.withLogGroupId("8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxxx");
request.withBody(body);
try {
 CreateTransferResponse response = client.createTransfer(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- 创建DIS转储

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateTransferSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 }
}
```

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
CreateTransferRequest request = new CreateTransferRequest();
CreateTransferRequestBody body = new CreateTransferRequestBody();
TransferDetail logTransferDetailLogTransferInfo = new TransferDetail();
logTransferDetailLogTransferInfo.withDisId("i1y8vfMTvf4LQzxxxx")
 .withDisName("xxxx");
CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer logAgencyTransferLogTransferInfo
= new CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer();
logAgencyTransferLogTransferInfo.withAgencyDomainId("1d26cc8c86a840e28a4f8d0d078xxxx")
 .withAgencyDomainName("paas_apm_z004xxxx_xx")
 .withAgencyName("test20210325")
 .withAgencyProjectId("2a473356cca5487f8373be891bfxxxx")
 .withBeAgencyDomainId("1d26cc8c86a840e28a4f8d0d078xxxx")
 .withBeAgencyProjectId("2a473356cca5487f8373be891bfxxxx");
CreateTransferRequestBodyLogTransferInfo logTransferInfobody = new
CreateTransferRequestBodyLogTransferInfo();
logTransferInfobody.withLogTransferType("DIS")
 .withLogTransferMode(CreateTransferRequestBodyLogTransferInfo.LogTransferModeEnum.from
Value("xxxx"))
 .withLogStorageFormat(CreateTransferRequestBodyLogTransferInfo.LogStorageFormatEnum.fr
omValue("XXX"))
 .withLogTransferStatus(CreateTransferRequestBodyLogTransferInfo.LogTransferStatusEnum.fro
mValue("XXXXX"))
 .withLogAgencyTransfer(logAgencyTransferLogTransferInfo)
 .withLogTransferDetail(logTransferDetailLogTransferInfo);
List<CreateTransferRequestBodyLogStreams> listbodyLogStreams = new ArrayList<>();
listbodyLogStreams.add(
 new CreateTransferRequestBodyLogStreams()
 .withLogStreamId("c776e1a7-8548-430a-afe5-xxxxxxxxxxxxx")
);
body.withLogTransferInfo(logTransferInfobody);
body.withLogStreams(listbodyLogStreams);
body.withLogGroupId("8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxxx");
request.withBody(body);
try {
 CreateTransferResponse response = client.createTransfer(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 创建OBS转储

```
coding: utf-8

import os
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudskdlts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudskdlts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateTransferRequest()
 logTransferDetailLogTransferInfo = TransferDetail(
 obs_period=2,
 obs_encrypted_id="1bd90032-1424-481f-8558-ba49854xxxxx",
 obs_prefix_name="xxxxx",
 obs_period_unit="min",
 obs_bucket_name="xxxxx",
 obs_dir_pre_fix_name="/xx",
 obs_time_zone="UTC+01:00",
 obs_time_zone_id="Africa/Lagos"
)
 logAgencyTransferLogTransferInfo =
 CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer(
 agency_domain_id="1d26cc8c86a840e28a4f8d0d078xxxxx",
 agency_domain_name="paas_apm_z004xxxxx_xx",
 agency_name="test20210325",
 agency_project_id="2a473356cca5487f8373be891bfxxxxx",
 be_agency_domain_id="1d26cc8c86a840e28a4f8d0d078xxxxx",
 be_agency_project_id="2a473356cca5487f8373be891bfxxxxx"
)
 logTransferInfobody = CreateTransferRequestBodyLogTransferInfo(
 log_transfer_type="OBS",
 log_transfer_mode="xxxxx",
 log_storage_format="XXX",
 log_transfer_status="XXXXX",
 log_agency_transfer=logAgencyTransferLogTransferInfo,
 log_transfer_detail=logTransferDetailLogTransferInfo
)
 listLogStreamsbody = [
 CreateTransferRequestBodyLogStreams(
 log_stream_id="c776e1a7-8548-430a-afe5-xxxxxxxxxxxxx"
)
]
 request.body = CreateTransferRequestBody(
 log_transfer_info=logTransferInfobody,
 log_streams=listLogStreamsbody,
 log_group_id="8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxxx"
)
 response = client.create_transfer(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## ● 创建DIS转储

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateTransferRequest()
 logTransferDetailLogTransferInfo = TransferDetail(
 dis_id="i1y8vfMTvf4LQzxxxx",
 dis_name="xxxx"
)
 logAgencyTransferLogTransferInfo =
 CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer(
 agency_domain_id="1d26cc8c86a840e28a4f8d0d078xxxx",
 agency_domain_name="paas_apm_z004xxxx_xx",
 agency_name="test20210325",
 agency_project_id="2a473356cca5487f8373be891bfxxxx",
 be_agency_domain_id="1d26cc8c86a840e28a4f8d0d078xxxx",
 be_agency_project_id="2a473356cca5487f8373be891bfxxxx"
)
 logTransferInfobody = CreateTransferRequestBodyLogTransferInfo(
 log_transfer_type="DIS",
 log_transfer_mode="xxxx",
 log_storage_format="XXX",
 log_transfer_status="XXXX",
 log_agency_transfer=logAgencyTransferLogTransferInfo,
 log_transfer_detail=logTransferDetailLogTransferInfo
)
 listLogStreamsbody = [
 CreateTransferRequestBodyLogStreams(
 log_stream_id="c776e1a7-8548-430a-afe5-xxxxxxxxxxxx"
)
]
 request.body = CreateTransferRequestBody(
 log_transfer_info=logTransferInfobody,
 log_streams=listLogStreamsbody,
 log_group_id="8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxx"
)
 response = client.create_transfer(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 创建OBS转储

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateTransferRequest{}
 obsEncryptedIdLogTransferDetail := "1bd90032-1424-481f-8558-ba49854xxxxx"
 obsPrefixNameLogTransferDetail := "xxxxx"
 obsDirPreFixNameLogTransferDetail := "/xx"
 obsTimeZoneLogTransferDetail := "UTC+01:00"
 obsTimeZoneldLogTransferDetail := "Africa/Lagos"
 logTransferDetailLogTransferInfo := &model.TransferDetail{
 ObsPeriod: model.GetTransferDetailObsPeriodEnum().E_2,
 ObsEncryptedId: &obsEncryptedIdLogTransferDetail,
 ObsPrefixName: &obsPrefixNameLogTransferDetail,
 ObsPeriodUnit: "min",
 ObsBucketName: "xxxxx",
 ObsDirPreFixName: &obsDirPreFixNameLogTransferDetail,
 ObsTimeZone: &obsTimeZoneLogTransferDetail,
 ObsTimeZoneld: &obsTimeZoneldLogTransferDetail,
 }
 logAgencyTransferLogTransferInfo :=
 &model.CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer{
 AgencyDomainId: "1d26cc8c86a840e28a4f8d0d078xxxxx",
 AgencyDomainName: "paas_apm_z004xxxx_xx",
 AgencyName: "test20210325",
 AgencyProjectId: "2a473356cca5487f8373be891bfxxxxx",
 BeAgencyDomainId: "1d26cc8c86a840e28a4f8d0d078xxxxx",
 BeAgencyProjectId: "2a473356cca5487f8373be891bfxxxxx",
 }
 logTransferInfobody := &model.CreateTransferRequestBodyLogTransferInfo{
 LogTransferType: "OBS",
 LogTransferMode:
 model.GetCreateTransferRequestBodyLogTransferInfoLogTransferModeEnum().XXXXX,
 LogStorageFormat:
 model.GetCreateTransferRequestBodyLogTransferInfoLogStorageFormatEnum().XXX,
 LogTransferStatus:
 model.GetCreateTransferRequestBodyLogTransferInfoLogTransferStatusEnum().XXXXX,
```

```
 LogAgencyTransfer: logAgencyTransferLogTransferInfo,
 LogTransferDetail: logTransferDetailLogTransferInfo,
 }
 var listLogStreamsbody = []model.CreateTransferRequestBodyLogStreams{
 {
 LogStreamId: "c776e1a7-8548-430a-afe5-xxxxxxxxxxxx",
 },
 }
 request.Body = &model.CreateTransferRequestBody{
 LogTransferInfo: logTransferInfobody,
 LogStreams: listLogStreamsbody,
 LogGroupId: "8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxx",
 }
 response, err := client.CreateTransfer(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

- 创建DIS转储

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateTransferRequest{}
 disIdLogTransferDetail := "i1y8vfMTvf4LQzxxxx"
 disNameLogTransferDetail := "xxxx"
 logTransferDetailLogTransferInfo := &model.TransferDetail{
 DisId: &disIdLogTransferDetail,
 DisName: &disNameLogTransferDetail,
 }
 logAgencyTransferLogTransferInfo :=
 &model.CreateTransferRequestBodyLogTransferInfoLogAgencyTransfer{
 AgencyDomainId: "1d26cc8c86a840e28a4f8d0d078xxxx",
 AgencyDomainName: "paas_apm_z004xxxx_xx",
 AgencyName: "test20210325",
 AgencyProjectId: "2a473356cca5487f8373be891bfxxxx",
 BeAgencyDomainId: "1d26cc8c86a840e28a4f8d0d078xxxx",
 BeAgencyProjectId: "2a473356cca5487f8373be891bfxxxx",
 }
```

```
}
logTransferInfoBody := &model.CreateTransferRequestBodyLogTransferInfo{
 LogTransferType: "DIS",
 LogTransferMode:
model.GetCreateTransferRequestBodyLogTransferInfoLogTransferModeEnum().XXXXX,
 LogStorageFormat:
model.GetCreateTransferRequestBodyLogTransferInfoLogStorageFormatEnum().XXX,
 LogTransferStatus:
model.GetCreateTransferRequestBodyLogTransferInfoLogTransferStatusEnum().XXXXX,
 LogAgencyTransfer: logAgencyTransferLogTransferInfo,
 LogTransferDetail: logTransferDetailLogTransferInfo,
}
var listLogStreamsbody = []model.CreateTransferRequestBodyLogStreams{
 {
 LogStreamId: "c776e1a7-8548-430a-afe5-xxxxxxxxxxxx",
 },
}
request.Body = &model.CreateTransferRequestBody{
 LogTransferInfo: logTransferInfoBody,
 LogStreams: listLogStreamsbody,
 LogGroupId: "8ba9e43f-be60-4d8c-9015-xxxxxxxxxxxx",
}
response, err := client.CreateTransfer(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                         |
|-----|--------------------------------------------|
| 200 | 创建转储请求响应成功。                                |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。      |
| 500 | InternalServerError。表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

### 6.6.3 删除日志转储

#### 功能介绍

该接口用于删除OBS转储，DIS转储，DMS转储。

#### 调用方法

请参见[如何调用API](#)。

## URI

DELETE /v2/{project\_id}/transfers

表 6-324 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-325 Query 参数

| 参数              | 是否必选 | 参数类型   | 描述                                                                                       |
|-----------------|------|--------|------------------------------------------------------------------------------------------|
| log_transfer_id | 是    | String | 日志转储ID。获取ID有3种方式：1. 调用查询日志转储接口，返回值有日志转储ID 2. 调用新增日志转储接口，返回值有日志转储ID<br>最小长度：36<br>最大长度：36 |

## 请求参数

表 6-326 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-327 响应 Body 参数

| 参数                | 参数类型                                              | 描述                           |
|-------------------|---------------------------------------------------|------------------------------|
| log_group_id      | String                                            | 日志组ID<br>最小长度：36<br>最大长度：36  |
| log_group_name    | String                                            | 日志组名称<br>最小长度：1<br>最大长度：64   |
| log_streams       | Array of <a href="#">log_streams</a> objects      | 日志流集合                        |
| log_transfer_id   | String                                            | 日志转储ID<br>最小长度：36<br>最大长度：36 |
| log_transfer_info | <a href="#">log_transfer_info_RespBody</a> object | 日志转储信息                       |

表 6-328 log\_streams

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：64  |

表 6-329 log\_transfer\_info\_RespBody

| 参数                   | 参数类型                                        | 描述                                  |
|----------------------|---------------------------------------------|-------------------------------------|
| log_agency_transf er | <a href="#">log_agency_transf er</a> object | 委托转储信息。若转储为委托转储，则会返回该参数             |
| log_create_time      | Integer                                     | 日志转储创建时间<br>最小值：0<br>最大值：2147483647 |

| 参数                  | 参数类型                         | 描述                                                                                                                                                                                                  |
|---------------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_storage_format  | String                       | 日志转储格式。只支持"RAW", "JSON"。RAW是指原始日志格式, JSON是指JSON日志格式。OBS转储和DIS转储支持JSON和RAW, DMS转储仅支持RAW<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>JSON</b></li> <li>• <b>RAW</b></li> </ul>             |
| log_transfer_detail | <b>TransferDetail</b> object | 日志转储详细信息                                                                                                                                                                                            |
| log_transfer_mode   | String                       | 日志转储方式。cycle是指周期性转储, realTime是指实时转储。OBS转储只支持"cycle", DIS转储和DMS转储只支持"realTime"。<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>cycle</b></li> <li>• <b>realTime</b></li> </ul>               |
| log_transfer_status | String                       | 日志转储状态, ENABLE是指日志转储开启状态, DISABLE是指日志转储关闭状态, EXCEPTION是指日志转储异常状态<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>ENABLE</b></li> <li>• <b>DISABLE</b></li> <li>• <b>EXCEPTION</b></li> </ul> |
| log_transfer_type   | String                       | 日志转储类型。OBS指OBS日志转储, DIS指DIS日志转储, DMS指DMS日志转储。<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>OBS</b></li> <li>• <b>DIS</b></li> <li>• <b>DMS</b></li> </ul>                                 |

表 6-330 log\_agency\_transfer

| 参数               | 参数类型   | 描述                              |
|------------------|--------|---------------------------------|
| agency_domain_id | String | 委托方账号ID<br>最小长度: 1<br>最大长度: 128 |

| 参数                   | 参数类型   | 描述                                            |
|----------------------|--------|-----------------------------------------------|
| agency_domain_name   | String | 委托方账号名称<br>最小长度：1<br>最大长度：128                 |
| agency_name          | String | 委托方配置的委托名称<br>最小长度：1<br>最大长度：128              |
| agency_project_id    | String | 委托方项目ID<br>最小长度：32<br>最大长度：32                 |
| be_agency_domain_id  | String | 被委托方账号ID，实际配置转储的账号ID<br>最小长度：1<br>最大长度：128    |
| be_agency_project_id | String | 被委托方项目ID，实际配置转储的账号的项目ID<br>最小长度：32<br>最大长度：32 |

表 6-331 TransferDetail

| 参数         | 参数类型    | 描述                                                                                                                                                                                                                                                                                           |
|------------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> <li>• 3</li> <li>• 5</li> <li>• 6</li> <li>• 12</li> <li>• 30</li> </ul> |

| 参数                   | 参数类型   | 描述                                                                                                                                                                                                                                         |
|----------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period_unit      | String | <p>OBS转储单位。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。</p> <p>枚举值：</p> <ul style="list-style-type: none"> <li>• “min”</li> <li>• “hour”</li> </ul> |
| obs_bucket_name      | String | <p>OBS日志桶名称。当创建OBS转储时，必填此参数。</p> <p>最小长度：3<br/>最大长度：63</p> <p>最小长度：3<br/>最大长度：63</p>                                                                                                                                                       |
| obs_encrypted_id     | String | <p>OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志桶加密则必须填写该参数，若OBS转储日志桶则不需要此参数。</p> <p>最小长度：36<br/>最大长度：36</p> <p>最小长度：36<br/>最大长度：36</p>                                                                                                           |
| obs_dir_pre_fix_name | String | <p>OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。</p> <p>正则约束：<br/>^(/)?([a-zA-Z0-9.-]+)/([a-zA-Z0-9.-]+)?/?\$</p> <p>最小长度：1<br/>最大长度：64</p> <p>最小长度：1<br/>最大长度：64</p>                                                                                  |
| obs_prefix_name      | String | <p>OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。</p> <p>正则约束：<br/>^[a-zA-Z0-9._]*\$</p> <p>最小长度：1<br/>最大长度：64</p> <p>最小长度：1<br/>最大长度：64</p>                                                                                                             |

| 参数                | 参数类型   | 描述                                                                                                                 |
|-------------------|--------|--------------------------------------------------------------------------------------------------------------------|
| obs_time_zone     | String | OBS转储时区，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone_id。                                           |
| obs_time_zone_id  | String | OBS转储时区ID，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone。                                            |
| dis_id            | String | DIS转储通道ID。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                             |
| dis_name          | String | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                             |
| kafka_id          | String | DMS转储kafka ID。当创建DMS转储时，必填此参数。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：36<br>最大长度：36 |
| kafka_topic       | String | DMS转储kafka topic。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：1<br>最大长度：128              |
| obs_transfer_path | String | OBS转储路径，指OBS日志桶中的路径。<br>最小长度：0<br>最大长度：256                                                                         |
| obs_eps_id        | String | OBS企业项目ID。<br>最小长度：0<br>最大长度：128                                                                                   |

| 参数                 | 参数类型             | 描述                                                                                                                                                       |
|--------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_encrypted_able | Boolean          | OBS是否开启加密。<br>枚举值：<br><ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul>                                                          |
| tags               | Array of strings | 若开启tag投递，该字段必须包含主机信息：hostIP、hostId、hostName、pathFile、collectTime；<br>公共字段有：logStreamName、regionName、logGroupName、projectId，为可选填；<br>开启转储标签：streamTag，可选填 |

状态码：400

表 6-332 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-333 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

根据日志转储ID，删除日志转储

```
DELETE https://{endpoint}/v2/{project_id}/transfers
/v2/{project_id}/transfers?log_transfer_id=cfc43c45-9edc-4a03-8578-0eb00cxxxxx
```

## 响应示例

### 状态码：200

删除转储请求响应成功。

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "obs_period": 2,
 "obs_prefix_name": "",
 "obs_period_unit": "min",
 "obs_transfer_path": "/0002/LogTanks/xxx/",
 "obs_bucket_name": "0002",
 "obs_encrypted_enable": false,
 "obs_dir_pre_fix_name": "",
 "obs_time_zone": "UTC+01:00",
 "obs_time_zone_id": "Africa/Lagos",
 "dis_id": "xxxxx",
 "dis_name": "xxxxxx",
 "kafka_id": "xxxxxx",
 "kafka_topic": "xxxxx"
 },
 "log_transfer_mode": "cycle",
 "log_transfer_status": "ENABLE",
 "log_transfer_type": "OBS"
 }
}
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.0405",
 "error_msg": "The log transfer does not existed"
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0010",
 "error_msg": "The system encountered an internal error"
}
```

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 删除转储请求响应成功。                           |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.6.4 更新日志转储

### 功能介绍

该接口用于更新OBS转储，DIS转储，DMS转储。

### 调用方法

请参见[如何调用API](#)。

## URI

PUT /v2/{project\_id}/transfers

表 6-334 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-335 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-336 请求 Body 参数

| 参数                | 是否必选 | 参数类型                                            | 描述                           |
|-------------------|------|-------------------------------------------------|------------------------------|
| log_transfer_id   | 是    | String                                          | 日志转储ID<br>最小长度：36<br>最大长度：36 |
| log_transfer_info | 是    | <a href="#">update_log_transfer_info</a> object | 日志转储信息                       |

表 6-337 update\_log\_transfer\_info

| 参数                 | 是否必选 | 参数类型   | 描述                                                                                                                              |
|--------------------|------|--------|---------------------------------------------------------------------------------------------------------------------------------|
| log_storage_format | 是    | String | 日志转储格式。只支持"RAW", "JSON"。RAW是指原始日志格式，JSON是指JSON日志格式。<br>枚举值： <ul style="list-style-type: none"> <li>JSON</li> <li>RAW</li> </ul> |

| 参数                  | 是否必选 | 参数类型                              | 描述                                                                                                                                                                                        |
|---------------------|------|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_status | 是    | String                            | 日志转储状态，ENABLE是指日志转储开启状态，DISABLE是指日志转储关闭状态，EXCEPTION是指日志转储异常状态<br>枚举值： <ul style="list-style-type: none"><li>• <b>ENABLE</b></li><li>• <b>DISABLE</b></li><li>• <b>EXCEPTION</b></li></ul> |
| log_transfer_detail | 是    | <b>log_transfer_detail</b> object | 日志转储详细信息                                                                                                                                                                                  |

表 6-338 log\_transfer\_detail

| 参数              | 是否必选 | 参数类型    | 描述                                                                                                                                                                                                                                                                                                                                 |
|-----------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period      | 是    | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• <b>1</b></li><li>• <b>2</b></li><li>• <b>3</b></li><li>• <b>5</b></li><li>• <b>6</b></li><li>• <b>12</b></li><li>• <b>30</b></li></ul> |
| obs_period_unit | 是    | String  | OBS转储单位。当创建OBS转储时，必填此参数。与obs_period组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>• <b>min</b></li><li>• <b>hour</b></li></ul>                                                                                                  |

| 参数                  | 是否必选 | 参数类型   | 描述                                                                                       |
|---------------------|------|--------|------------------------------------------------------------------------------------------|
| obs_bucket_name     | 是    | String | OBS转储日志桶名称。当创建OBS转储时，必填此参数。<br>最小长度：3<br>最大长度：63                                         |
| obs_encrypted_id    | 否    | String | OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志加密桶则必须填写该参数，若OBS转储日志桶则不需要此参数<br>最小长度：36<br>最大长度：36 |
| obs_dir_prefix_name | 否    | String | OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。<br>最小长度：1<br>最大长度：64                                   |
| obs_prefix_name     | 否    | String | OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。<br>最小长度：1<br>最大长度：64                                    |
| obs_time_zone       | 否    | String | OBS转储时区。参数选择参考OBS转储时区表。如果选择该参数，则必须选择obs_time_zone_id。                                    |
| obs_time_zone_id    | 否    | String | OBS转储时区ID。参数选择参考OBS转储时区表。如果选择该参数，则必须选择obs_time_zone。                                     |
| dis_id              | 否    | String | DIS转储通道ID。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128                                         |
| dis_name            | 否    | String | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：64                                          |

| 参数                   | 是否必选 | 参数类型             | 描述                                                                                                                                                           |
|----------------------|------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| kafka_id             | 否    | String           | DMS转储kafka ID。当创建DMS转储时，必填此参数。创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例                                                                     |
| kafka_topic          | 否    | String           | DMS转储kafka topic。当创建DMS转储时，必填此参数。创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例                                                                  |
| obs_eps_id           | 否    | String           | OBS企业项目ID。                                                                                                                                                   |
| obs_encrypted_enable | 否    | Boolean          | OBS是否开启加密。                                                                                                                                                   |
| tags                 | 否    | Array of strings | 若开启tag投递，该字段必须包含主机信息：hostIP、hostId、hostName、pathFile、collectTime；<br>公共字段有：<br>logStreamName、regionName、logGroupName、projectId，为可选填；开启转储标签：<br>streamTag，可选填 |
| lts_tags             | 否    | Array of strings | dms转储JSON格式选填，可以转储tag字段                                                                                                                                      |
| stream_tags          | 否    | Array of strings | dms转储JSON格式选填，可以转储日志流标签字段                                                                                                                                    |
| struct_fields        | 否    | Array of strings | dms转储JSON格式选填，可以转储结构化字段                                                                                                                                      |
| invalid_field_value  | 否    | String           | dms转储JSON格式选填，无效字段填充                                                                                                                                         |

## 响应参数

状态码：200

表 6-339 响应 Body 参数

| 参数                | 参数类型                                              | 描述                           |
|-------------------|---------------------------------------------------|------------------------------|
| log_group_id      | String                                            | 日志组ID<br>最小长度：36<br>最大长度：36  |
| log_group_name    | String                                            | 日志组名称<br>最小长度：1<br>最大长度：64   |
| log_streams       | Array of <a href="#">log_streams</a> objects      | 日志流集合                        |
| log_transfer_id   | String                                            | 日志转储ID<br>最小长度：36<br>最大长度：36 |
| log_transfer_info | <a href="#">log_transfer_info_RespBody</a> object | 日志转储信息                       |

表 6-340 log\_streams

| 参数              | 参数类型   | 描述                          |
|-----------------|--------|-----------------------------|
| log_stream_id   | String | 日志流ID<br>最小长度：36<br>最大长度：36 |
| log_stream_name | String | 日志流名称<br>最小长度：1<br>最大长度：64  |

表 6-341 log\_transfer\_info\_RespBody

| 参数                   | 参数类型                                        | 描述                                  |
|----------------------|---------------------------------------------|-------------------------------------|
| log_agency_transf er | <a href="#">log_agency_trans fer</a> object | 委托转储信息。若转储为委托转储，则会返回该参数             |
| log_create_time      | Integer                                     | 日志转储创建时间<br>最小值：0<br>最大值：2147483647 |

| 参数                  | 参数类型                         | 描述                                                                                                                                                                                               |
|---------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_storage_format  | String                       | 日志转储格式。只支持"RAW", "JSON"。RAW是指原始日志格式，JSON是指JSON日志格式。OBS转储和DIS转储支持JSON和RAW，DMS转储仅支持RAW<br>枚举值：<br><ul style="list-style-type: none"> <li>• <b>JSON</b></li> <li>• <b>RAW</b></li> </ul>            |
| log_transfer_detail | <b>TransferDetail</b> object | 日志转储详细信息                                                                                                                                                                                         |
| log_transfer_mode   | String                       | 日志转储方式。cycle是指周期性转储，realTime是指实时转储。OBS转储只支持"cycle"，DIS转储和DMS转储只支持"realTime"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• <b>cycle</b></li> <li>• <b>realTime</b></li> </ul>              |
| log_transfer_status | String                       | 日志转储状态，ENABLE是指日志转储开启状态，DISABLE是指日志转储关闭状态，EXCEPTION是指日志转储异常状态<br>枚举值：<br><ul style="list-style-type: none"> <li>• <b>ENABLE</b></li> <li>• <b>DISABLE</b></li> <li>• <b>EXCEPTION</b></li> </ul> |
| log_transfer_type   | String                       | 日志转储类型。OBS指OBS日志转储，DIS指DIS日志转储，DMS指DMS日志转储。<br>枚举值：<br><ul style="list-style-type: none"> <li>• <b>OBS</b></li> <li>• <b>DIS</b></li> <li>• <b>DMS</b></li> </ul>                                |

表 6-342 log\_agency\_transfer

| 参数               | 参数类型   | 描述                            |
|------------------|--------|-------------------------------|
| agency_domain_id | String | 委托方账号ID<br>最小长度：1<br>最大长度：128 |

| 参数                   | 参数类型   | 描述                                            |
|----------------------|--------|-----------------------------------------------|
| agency_domain_name   | String | 委托方账号名称<br>最小长度：1<br>最大长度：128                 |
| agency_name          | String | 委托方配置的委托名称<br>最小长度：1<br>最大长度：128              |
| agency_project_id    | String | 委托方项目ID<br>最小长度：32<br>最大长度：32                 |
| be_agency_domain_id  | String | 被委托方账号ID，实际配置转储的账号ID<br>最小长度：1<br>最大长度：128    |
| be_agency_project_id | String | 被委托方项目ID，实际配置转储的账号的项目ID<br>最小长度：32<br>最大长度：32 |

表 6-343 TransferDetail

| 参数         | 参数类型    | 描述                                                                                                                                                                                                                                                                                           |
|------------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> <li>• 3</li> <li>• 5</li> <li>• 6</li> <li>• 12</li> <li>• 30</li> </ul> |

| 参数                   | 参数类型   | 描述                                                                                                                                                                                                                       |
|----------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period_unit      | String | OBS转储单位。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值： <ul style="list-style-type: none"><li>“min”</li><li>“hour”</li></ul> |
| obs_bucket_name      | String | OBS日志桶名称。当创建OBS转储时，必填此参数。<br>最小长度：3<br>最大长度：63<br>最小长度：3<br>最大长度：63                                                                                                                                                      |
| obs_encrypted_id     | String | OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志桶加密则必须填写该参数，若OBS转储日志桶则不需要此参数。<br>最小长度：36<br>最大长度：36<br>最小长度：36<br>最大长度：36                                                                                                          |
| obs_dir_pre_fix_name | String | OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。<br>正则约束：<br>^(/)?([a-zA-Z0-9.-]+)(/[a-zA-Z0-9.-]+)*(/)?\$<br>最小长度：1<br>最大长度：64<br>最小长度：1<br>最大长度：64                                                                                    |
| obs_prefix_name      | String | OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。<br>正则约束：<br>^[a-zA-Z0-9._]*\$<br>最小长度：1<br>最大长度：64<br>最小长度：1<br>最大长度：64                                                                                                                 |

| 参数                | 参数类型   | 描述                                                                                                                 |
|-------------------|--------|--------------------------------------------------------------------------------------------------------------------|
| obs_time_zone     | String | OBS转储时区，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone_id。                                           |
| obs_time_zone_id  | String | OBS转储时区ID，参数选择请参考 <a href="#">OBS转储时区表</a> 。如果选择该参数，则必须选择obs_time_zone。                                            |
| dis_id            | String | DIS转储通道ID。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                             |
| dis_name          | String | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                             |
| kafka_id          | String | DMS转储kafka ID。当创建DMS转储时，必填此参数。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：36<br>最大长度：36 |
| kafka_topic       | String | DMS转储kafka topic。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：1<br>最大长度：128              |
| obs_transfer_path | String | OBS转储路径，指OBS日志桶中的路径。<br>最小长度：0<br>最大长度：256                                                                         |
| obs_eps_id        | String | OBS企业项目ID。<br>最小长度：0<br>最大长度：128                                                                                   |

| 参数                 | 参数类型             | 描述                                                                                                                                                       |
|--------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_encrypted_able | Boolean          | OBS是否开启加密。<br>枚举值： <ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>                                                                |
| tags               | Array of strings | 若开启tag投递，该字段必须包含主机信息：hostIP、hostId、hostName、pathFile、collectTime；<br>公共字段有：logStreamName、regionName、logGroupName、projectId，为可选填；<br>开启转储标签：streamTag，可选填 |

状态码：400

表 6-344 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-345 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

- 更新OBS转储

```
PUT https://{endpoint}/v2/{project_id}/transfers
```

```
{
 "log_transfer_id": "9f74e101-b969-483c-a610-d3f3064xxxxx",
 "log_transfer_info": {
```

```
"log_storage_format": "JSON",
"log_transfer_status": "DISABLE",
"log_transfer_detail": {
 "obs_period": 3,
 "obs_period_unit": "hour",
 "obs_bucket_name": "0xxx",
 "obs_encrypted_id": "1bd90032-1424-481f-8558-ba49854xxxxx",
 "obs_dir_pre_fix_name": "/xx",
 "obs_prefix_name": "xxxxx",
 "obs_time_zone": "UTC+01:00",
 "obs_time_zone_id": "Africa/Lagos"
}
}
```

- 更新DIS转储

PUT https://{endpoint}/v2/{project\_id}/transfers

```
{
 "log_transfer_id": "9f74e101-b969-483c-a610-d3f3064xxxxx",
 "log_transfer_info": {
 "log_storage_format": "JSON",
 "log_transfer_status": "DISABLE",
 "log_transfer_detail": {
 "dis_id": "xxxxx",
 "dis_name": "xxxxxx"
 }
 }
}
```

- 更新DMS转储

PUT https://{endpoint}/v2/{project\_id}/transfers

```
{
 "log_transfer_id": "9f74e101-b969-483c-a610-d3f3064xxxxx",
 "log_transfer_info": {
 "log_storage_format": "JSON",
 "log_transfer_status": "DISABLE",
 "log_transfer_detail": {
 "kafka_id": "xxxxx",
 "kafka_topic": "xxxxxx"
 }
 }
}
```

## 响应示例

### 状态码：200

更新转储请求响应成功。

- 当创建OBS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 }
 }
}
```

```
"agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
"be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
"be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
},
"log_transfer_detail": {
 "obs_period": 2,
 "obs_prefix_name": "",
 "obs_period_unit": "min",
 "obs_transfer_path": "/0002/LogTanks/cn-north-7/",
 "obs_bucket_name": "0002",
 "obs_encrypted_enable": false,
 "obs_dir_pre_fix_name": "",
 "obs_time_zone": "UTC+01:00",
 "obs_time_zone_id": "Africa/Lagos",
 "tags": []
},
"log_transfer_mode": "cycle",
"log_transfer_status": "ENABLE",
"log_transfer_type": "OBS"
}
}
```

- 当创建DIS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail": {
 "dis_id": "xxxxx",
 "dis_name": "xxxxx",
 "tags": []
 },
 "log_transfer_mode": "cycle",
 "log_transfer_status": "ENABLE",
 "log_transfer_type": "OBS"
 }
}
```

- 当创建DMS转储时，会返回如下参数

```
{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lts-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lts-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
 "log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",

```

```
"be_agency_domain_id" : "1d26cc8c86a840e28a4f8d0d078xxxx",
"be_agency_project_id" : "2a473356cca5487f8373be891bfxxxx"
},
"log_transfer_detail" : {
 "kafka_id" : "xxxxx",
 "kafka_topic" : "xxxxx",
 "tags" : []
},
"log_transfer_mode" : "cycle",
"log_transfer_status" : "ENABLE",
"log_transfer_type" : "OBS"
}
}
```

**状态码：400**

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.0009",
 "error_msg" : "The Field transfer id is invalid or missing."
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0010",
 "error_msg" : "The system encountered an internal error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 更新OBS转储

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateTransferSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
```

```
.withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateTransferRequest request = new UpdateTransferRequest();
UpdateTransferRequestBody body = new UpdateTransferRequestBody();
TransferDetail logTransferDetailLogTransferInfo = new TransferDetail();
logTransferDetailLogTransferInfo.withObsPeriod(TransferDetail.ObsPeriodEnum.NUMBER_3)
 .withObsEncryptedId("1bd90032-1424-481f-8558-ba49854xxxxx")
 .withObsPrefixName("xxxxx")
 .withObsPeriodUnit("hour")
 .withObsBucketName("0xxx")
 .withObsDirPreFixName("/xx")
 .withObsTimeZone("UTC+01:00")
 .withObsTimeZoneld("Africa/Lagos");
UpdateTransferRequestBodyLogTransferInfo logTransferInfobody = new
UpdateTransferRequestBodyLogTransferInfo();

logTransferInfobody.withLogStorageFormat(UpdateTransferRequestBodyLogTransferInfo.LogStorageFormatEnum.fromValue("JSON"))
 .withLogTransferStatus(UpdateTransferRequestBodyLogTransferInfo.LogTransferStatusEnum.fromValue("DISABLE"))
 .withLogTransferDetail(logTransferDetailLogTransferInfo);
body.withLogTransferInfo(logTransferInfobody);
body.withLogTransferId("9f74e101-b969-483c-a610-d3f3064xxxxx");
request.withBody(body);
try {
 UpdateTransferResponse response = client.updateTransfer(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- 更新DIS转储

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateTransferSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateTransferRequest request = new UpdateTransferRequest();
UpdateTransferRequestBody body = new UpdateTransferRequestBody();
TransferDetail logTransferDetailLogTransferInfo = new TransferDetail();
logTransferDetailLogTransferInfo.withDisId("xxxxx")
 .withDisName("xxxxxx");
UpdateTransferRequestBodyLogTransferInfo logTransferInfobody = new
UpdateTransferRequestBodyLogTransferInfo();

logTransferInfobody.withLogStorageFormat(UpdateTransferRequestBodyLogTransferInfo.LogStorageFor
matEnum.fromValue("JSON"))
 .withLogTransferStatus(UpdateTransferRequestBodyLogTransferInfo.LogTransferStatusEnum.fro
mValue("DISABLE"))
 .withLogTransferDetail(logTransferDetailLogTransferInfo);
body.withLogTransferInfo(logTransferInfobody);
body.withLogTransferId("9f74e101-b969-483c-a610-d3f3064xxxxx");
request.withBody(body);
try {
 UpdateTransferResponse response = client.updateTransfer(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- **更新DMS转储**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateTransferSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
```

```
.withProjectId(projectId)
.withAk(ak)
.withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateTransferRequest request = new UpdateTransferRequest();
UpdateTransferRequestBody body = new UpdateTransferRequestBody();
TransferDetail logTransferDetailLogTransferInfo = new TransferDetail();
logTransferDetailLogTransferInfo.withKafkaId("xxxxx")
 .withKafkaTopic("xxxxx");
UpdateTransferRequestBodyLogTransferInfo logTransferInfobody = new
UpdateTransferRequestBodyLogTransferInfo();

logTransferInfobody.withLogStorageFormat(UpdateTransferRequestBodyLogTransferInfo.LogStorageFor
matEnum.fromValue("JSON"))
 .withLogTransferStatus(UpdateTransferRequestBodyLogTransferInfo.LogTransferStatusEnum.fro
mValue("DISABLE"))
 .withLogTransferDetail(logTransferDetailLogTransferInfo);
body.withLogTransferInfo(logTransferInfobody);
body.withLogTransferId("9f74e101-b969-483c-a610-d3f3064xxxxx");
request.withBody(body);
try {
 UpdateTransferResponse response = client.updateTransfer(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 更新OBS转储

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()
```

```
try:
 request = UpdateTransferRequest()
 logTransferDetailLogTransferInfo = TransferDetail(
 obs_period=3,
 obs_encrypted_id="1bd90032-1424-481f-8558-ba49854xxxxx",
 obs_prefix_name="xxxxx",
 obs_period_unit="hour",
 obs_bucket_name="Oxxx",
 obs_dir_pre_fix_name="/xx",
 obs_time_zone="UTC+01:00",
 obs_time_zone_id="Africa/Lagos"
)
 logTransferInfobody = UpdateTransferRequestBodyLogTransferInfo(
 log_storage_format="JSON",
 log_transfer_status="DISABLE",
 log_transfer_detail=logTransferDetailLogTransferInfo
)
 request.body = UpdateTransferRequestBody(
 log_transfer_info=logTransferInfobody,
 log_transfer_id="9f74e101-b969-483c-a610-d3f3064xxxxx"
)
 response = client.update_transfer(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- **更新DIS转储**

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateTransferRequest()
 logTransferDetailLogTransferInfo = TransferDetail(
 dis_id="xxxxx",
 dis_name="xxxxxx"
)
 logTransferInfobody = UpdateTransferRequestBodyLogTransferInfo(
 log_storage_format="JSON",
 log_transfer_status="DISABLE",
 log_transfer_detail=logTransferDetailLogTransferInfo
)
 request.body = UpdateTransferRequestBody(
 log_transfer_info=logTransferInfobody,
```

```
 log_transfer_id="9f74e101-b969-483c-a610-d3f3064xxxxx"
)
 response = client.update_transfer(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 更新DMS转储

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateTransferRequest()
 logTransferDetailLogTransferInfo = TransferDetail(
 kafka_id="xxxxx",
 kafka_topic="xxxxxx"
)
 logTransferInfobody = UpdateTransferRequestBodyLogTransferInfo(
 log_storage_format="JSON",
 log_transfer_status="DISABLE",
 log_transfer_detail=logTransferDetailLogTransferInfo
)
 request.body = UpdateTransferRequestBody(
 log_transfer_info=logTransferInfobody,
 log_transfer_id="9f74e101-b969-483c-a610-d3f3064xxxxx"
)
 response = client.update_transfer(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 更新OBS转储

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateTransferRequest{
 obsEncryptedIdLogTransferDetail:= "1bd90032-1424-481f-8558-ba49854xxxxx"
 obsPrefixNameLogTransferDetail:= "xxxxx"
 obsDirPreFixNameLogTransferDetail:= "/xx"
 obsTimeZoneLogTransferDetail:= "UTC+01:00"
 obsTimeZoneldLogTransferDetail:= "Africa/Lagos"
 logTransferDetailLogTransferInfo := &model.TransferDetail{
 ObsPeriod: model.GetTransferDetailObsPeriodEnum().E_3,
 ObsEncryptedId: &obsEncryptedIdLogTransferDetail,
 ObsPrefixName: &obsPrefixNameLogTransferDetail,
 ObsPeriodUnit: "hour",
 ObsBucketName: "0xxx",
 ObsDirPreFixName: &obsDirPreFixNameLogTransferDetail,
 ObsTimeZone: &obsTimeZoneLogTransferDetail,
 ObsTimeZoneld: &obsTimeZoneldLogTransferDetail,
 }
 logTransferInfobody := &model.UpdateTransferRequestBodyLogTransferInfo{
 LogStorageFormat:
model.GetUpdateTransferRequestBodyLogTransferInfoLogStorageFormatEnum().JSON,
 LogTransferStatus:
model.GetUpdateTransferRequestBodyLogTransferInfoLogTransferStatusEnum().DISABLE,
 LogTransferDetail: logTransferDetailLogTransferInfo,
 }
 request.Body = &model.UpdateTransferRequestBody{
 LogTransferInfo: logTransferInfobody,
 LogTransferId: "9f74e101-b969-483c-a610-d3f3064xxxxx",
 }
 response, err := client.UpdateTransfer(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
 }
}
```

- 更新DIS转储

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateTransferRequest{}
 disIdLogTransferDetail := "xxxxx"
 disNameLogTransferDetail := "xxxxxx"
 logTransferDetailLogTransferInfo := &model.TransferDetail{
 DisId: &disIdLogTransferDetail,
 DisName: &disNameLogTransferDetail,
 }
 logTransferInfoBody := &model.UpdateTransferRequestBodyLogTransferInfo{
 LogStorageFormat:
model.GetUpdateTransferRequestBodyLogTransferInfoLogStorageFormatEnum().JSON,
 LogTransferStatus:
model.GetUpdateTransferRequestBodyLogTransferInfoLogTransferStatusEnum().DISABLE,
 LogTransferDetail: logTransferDetailLogTransferInfo,
 }
 request.Body = &model.UpdateTransferRequestBody{
 LogTransferInfo: logTransferInfoBody,
 LogTransferId: "9f74e101-b969-483c-a610-d3f3064xxxxx",
 }
 response, err := client.UpdateTransfer(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
}
```

- **更新DMS转储**

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

```
environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.UpdateTransferRequest{
 kafkaIdLogTransferDetail:= "xxxxx"
 kafkaTopicLogTransferDetail:= "xxxxxx"
 logTransferDetailLogTransferInfo := &model.TransferDetail{
 KafkaId: &kafkaIdLogTransferDetail,
 KafkaTopic: &kafkaTopicLogTransferDetail,
 }
 logTransferInfobody := &model.UpdateTransferRequestBodyLogTransferInfo{
 LogStorageFormat:
model.GetUpdateTransferRequestBodyLogTransferInfoLogStorageFormatEnum().JSON,
 LogTransferStatus:
model.GetUpdateTransferRequestBodyLogTransferInfoLogTransferStatusEnum().DISABLE,
 LogTransferDetail: logTransferDetailLogTransferInfo,
 }
 request.Body = &model.UpdateTransferRequestBody{
 LogTransferInfo: logTransferInfobody,
 LogTransferId: "9f74e101-b969-483c-a610-d3f3064xxxxx",
 }
}
response, err := client.UpdateTransfer(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 更新转储请求响应成功。                           |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.6.5 查询日志转储

### 功能介绍

该接口用于查询OBS转储，DIS转储，DMS转储配置。

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/transfers

表 6-346 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-347 Query 参数

| 参数                | 是否必选 | 参数类型   | 描述                                                                                                                                   |
|-------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_type | 否    | String | 日志转储类型。OBS指OBS日志转储，DIS指DIS日志转储，DMS指DMS日志转储<br>枚举值： <ul style="list-style-type: none"><li>• OBS</li><li>• DIS</li><li>• DMS</li></ul> |
| log_group_name    | 否    | String | 日志组名称<br>最小长度：1<br>最大长度：64                                                                                                           |
| log_stream_name   | 否    | String | 日志流名称<br>最小长度：1<br>最大长度：64                                                                                                           |

## 请求参数

表 6-348 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-349 响应 Body 参数

| 参数            | 参数类型                                                        | 描述         |
|---------------|-------------------------------------------------------------|------------|
| log_transfers | Array of <a href="#">CreateTransferResponseBody</a> objects | 查询日志转储信息数组 |

表 6-350 CreateTransferResponseBody

| 参数             | 参数类型                                         | 描述                          |
|----------------|----------------------------------------------|-----------------------------|
| log_group_id   | String                                       | 日志组ID<br>最小长度：36<br>最大长度：36 |
| log_group_name | String                                       | 日志组名称<br>最小长度：1<br>最大长度：64  |
| log_streams    | Array of <a href="#">log_streams</a> objects | 日志流集合                       |

| 参数                | 参数类型                                              | 描述                                           |
|-------------------|---------------------------------------------------|----------------------------------------------|
| log_transfer_id   | String                                            | 日志转储ID<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| log_transfer_info | <a href="#">log_transfer_info_RespBody</a> object | 日志转储信息                                       |

表 6-351 log\_streams

| 参数              | 参数类型   | 描述                                          |
|-----------------|--------|---------------------------------------------|
| log_stream_id   | String | 日志流ID<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| log_stream_name | String | 日志流名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>  |

表 6-352 log\_transfer\_info\_RespBody

| 参数                   | 参数类型                                        | 描述                                                                                                                                                                                      |
|----------------------|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_agency_transf er | <a href="#">log_agency_transf er</a> object | 委托转储信息。若转储为委托转储, 则会返回该参数                                                                                                                                                                |
| log_create_time      | Integer                                     | 日志转储创建时间<br>最小值: <b>0</b><br>最大值: <b>2147483647</b>                                                                                                                                     |
| log_storage_form at  | String                                      | 日志转储格式。只支持"RAW", "JSON"。RAW是指原始日志格式, JSON是指JSON日志格式。OBS转储和DIS转储支持JSON和RAW, DMS转储仅支持RAW<br>枚举值:<br><ul style="list-style-type: none"> <li>• <b>JSON</b></li> <li>• <b>RAW</b></li> </ul> |
| log_transfer_detail  | <a href="#">TransferDetail</a> object       | 日志转储详细信息                                                                                                                                                                                |

| 参数                  | 参数类型   | 描述                                                                                                                                                                          |
|---------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_transfer_mode   | String | 日志转储方式。cycle是指周期性转储，realTime是指实时转储。OBS转储只支持"cycle"，DIS转储和DMS转储只支持"realTime"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• cycle</li> <li>• realTime</li> </ul>       |
| log_transfer_status | String | 日志转储状态，ENABLE是指日志转储开启状态，DISABLE是指日志转储关闭状态，EXCEPTION是指日志转储异常状态<br>枚举值：<br><ul style="list-style-type: none"> <li>• ENABLE</li> <li>• DISABLE</li> <li>• EXCEPTION</li> </ul> |
| log_transfer_type   | String | 日志转储类型。OBS指OBS日志转储，DIS指DIS日志转储，DMS指DMS日志转储。<br>枚举值：<br><ul style="list-style-type: none"> <li>• OBS</li> <li>• DIS</li> <li>• DMS</li> </ul>                                |

表 6-353 log\_agency\_transfer

| 参数                 | 参数类型   | 描述                               |
|--------------------|--------|----------------------------------|
| agency_domain_id   | String | 委托方账号ID<br>最小长度：1<br>最大长度：128    |
| agency_domain_name | String | 委托方账号名称<br>最小长度：1<br>最大长度：128    |
| agency_name        | String | 委托方配置的委托名称<br>最小长度：1<br>最大长度：128 |
| agency_project_id  | String | 委托方项目ID<br>最小长度：32<br>最大长度：32    |

| 参数                   | 参数类型   | 描述                                            |
|----------------------|--------|-----------------------------------------------|
| be_agency_domain_id  | String | 被委托方账号ID，实际配置转储的账号ID<br>最小长度：1<br>最大长度：128    |
| be_agency_project_id | String | 被委托方项目ID，实际配置转储的账号的项目ID<br>最小长度：32<br>最大长度：32 |

表 6-354 TransferDetail

| 参数              | 参数类型    | 描述                                                                                                                                                                                                                                                                                           |
|-----------------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_period      | Integer | OBS转储时间。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> <li>• 3</li> <li>• 5</li> <li>• 6</li> <li>• 12</li> <li>• 30</li> </ul> |
| obs_period_unit | String  | OBS转储单位。当创建OBS转储时，必填此参数。与obs_period_unit组合，即"obs_period"+"obs_period_unit"，必须是"2min", "5min", "30min", "1hour", "3hour", "6hour", "12hour"。<br>枚举值：<br><ul style="list-style-type: none"> <li>• “min”</li> <li>• “hour”</li> </ul>                                                           |
| obs_bucket_name | String  | OBS日志桶名称。当创建OBS转储时，必填此参数。<br>最小长度：3<br>最大长度：63<br>最小长度：3<br>最大长度：63                                                                                                                                                                                                                          |

| 参数                   | 参数类型   | 描述                                                                                                                                                                    |
|----------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obs_encrypted_id     | String | <p>OBS转储KMS密钥ID。根据OBS转储日志桶是否加密判断，若OBS转储日志桶加密则必须填写该参数，若OBS转储日志桶则不需要此参数。</p> <p>最小长度：36<br/>最大长度：36<br/>最小长度：<b>36</b><br/>最大长度：<b>36</b></p>                           |
| obs_dir_pre_fix_name | String | <p>OBS转储自定义转储路径。当创建OBS转储时，根据需要选填此参数。</p> <p>正则约束：<br/>^(/)?([a-zA-Z0-9.-]+)/([a-zA-Z0-9-]+)*(/)?\$</p> <p>最小长度：1<br/>最大长度：64<br/>最小长度：<b>1</b><br/>最大长度：<b>64</b></p> |
| obs_prefix_name      | String | <p>OBS转储日志文件前缀。当创建OBS转储时，根据需要选填此参数。</p> <p>正则约束：<br/>^[a-zA-Z0-9._]*\$</p> <p>最小长度：1<br/>最大长度：64<br/>最小长度：<b>1</b><br/>最大长度：<b>64</b></p>                             |
| obs_time_zone        | String | <p>OBS转储时区，参数选择请参考<a href="#">OBS转储时区表</a>。如果选择该参数，则必须选择obs_time_zone_id。</p>                                                                                         |
| obs_time_zone_id     | String | <p>OBS转储时区ID，参数选择请参考<a href="#">OBS转储时区表</a>。如果选择该参数，则必须选择obs_time_zone。</p>                                                                                          |
| dis_id               | String | <p>DIS转储通道ID。当创建DIS转储时，必填此参数。</p> <p>最小长度：1<br/>最大长度：128<br/>最小长度：<b>1</b><br/>最大长度：<b>128</b></p>                                                                    |

| 参数                   | 参数类型             | 描述                                                                                                                                                       |
|----------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| dis_name             | String           | DIS转储通道名称。当创建DIS转储时，必填此参数。<br>最小长度：1<br>最大长度：128<br>最小长度：1<br>最大长度：128                                                                                   |
| kafka_id             | String           | DMS转储kafka ID。当创建DMS转储时，必填此参数。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：36<br>最大长度：36                                       |
| kafka_topic          | String           | DMS转储kafka topic。<br>创建DMS转储前，需要使用kafka ID以及kafka Topic进行实例注册。详情见接口注册DMSkafka实例<br>最小长度：1<br>最大长度：128                                                    |
| obs_transfer_path    | String           | OBS转储路径，指OBS日志桶中的路径。<br>最小长度：0<br>最大长度：256                                                                                                               |
| obs_eps_id           | String           | OBS企业项目ID。<br>最小长度：0<br>最大长度：128                                                                                                                         |
| obs_encrypted_enable | Boolean          | OBS是否开启加密。<br>枚举值： <ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>                                                                |
| tags                 | Array of strings | 若开启tag投递，该字段必须包含主机信息：hostIP、hostId、hostName、pathFile、collectTime；<br>公共字段有：logStreamName、regionName、logGroupName、projectId，为可选填；<br>开启转储标签：streamTag，可选填 |

状态码：400

表 6-355 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-356 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

若不传参数则查询所有日志转储信息。若根据log\_transfer\_type, log\_group\_name, log\_stream\_name这3中不同的参数, 则查询对应的日志转储。

GET https://{endpoint}/v2/{project\_id}/transfers

```
/v2/{project_id}/transfers /v2/{project_id}/transfers?log_group_name=lbs-group-txxx /v2/{project_id}/transfers?log_transfer_type=OBS /v2/{project_id}/transfers?log_stream_name=lbs-topic-testRxxx /v2/{project_id}/transfers?log_group_name=lbs-group-txxx&log_transfer_type=OBS /v2/{project_id}/transfers?log_group_name=lbs-group-txxx&log_stream_name=lbs-topic-testRxxx /v2/{project_id}/transfers?log_transfer_type=OBS&log_stream_name=lbs-topic-testRxxx /v2/{project_id}/transfers?log_group_name=lbs-group-txxx&log_transfer_type=OBS&log_stream_name=lbs-topic-testRxxx
```

## 响应示例

状态码：200

查询转储请求响应成功。

- 当查询OBS转储时, 会返回如下参数

```
{
 "log_transfers": [{
 "log_group_id": "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name": "lbs-group-kafka",
 "log_streams": [{
 "log_stream_id": "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name": "lbs-topic-kafka"
 }],
 "log_transfer_id": "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info": {
 "log_create_time": 1634802241847,
 "log_storage_format": "JSON",
```

```
"log_agency_transfer" : {
 "agency_domain_id" : "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name" : "paas_apm_z004xxxx_xx",
 "agency_name" : "test20210325",
 "agency_project_id" : "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id" : "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id" : "2a473356cca5487f8373be891bfxxxxx"
},
"log_transfer_detail" : {
 "obs_period" : 2,
 "obs_prefix_name" : "",
 "obs_period_unit" : "min",
 "obs_transfer_path" : "/0002/LogTanks/xxx/",
 "obs_bucket_name" : "0002",
 "obs_encrypted_enable" : false,
 "obs_dir_pre_fix_name" : "",
 "obs_time_zone" : "UTC+01:00",
 "obs_time_zone_id" : "Africa/Lagos"
},
"log_transfer_mode" : "cycle",
"log_transfer_status" : "ENABLE",
"log_transfer_type" : "OBS"
}
}]
}
```

- 当查询DIS转储时，会返回如下参数

```
{
 "log_transfers" : [{
 "log_group_id" : "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name" : "lts-group-kafka",
 "log_streams" : [{
 "log_stream_id" : "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name" : "lts-topic-kafka"
 }],
 "log_transfer_id" : "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info" : {
 "log_create_time" : 1634802241847,
 "log_storage_format" : "JSON",
 "log_agency_transfer" : {
 "agency_domain_id" : "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name" : "paas_apm_z004xxxx_xx",
 "agency_name" : "test20210325",
 "agency_project_id" : "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id" : "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id" : "2a473356cca5487f8373be891bfxxxxx"
 },
 "log_transfer_detail" : {
 "dis_id" : "xxxxx",
 "dis_name" : "xxxxx"
 },
 "log_transfer_mode" : "cycle",
 "log_transfer_status" : "ENABLE",
 "log_transfer_type" : "OBS"
 }
 }
]
```

- 当创建DMS转储时，会返回如下参数

```
{
 "log_transfers" : [{
 "log_group_id" : "9a7e2183-2d6d-4732-9a9b-e897fd4e49e0",
 "log_group_name" : "lts-group-kafka",
 "log_streams" : [{
 "log_stream_id" : "839dac89-35af-4db2-ab4a-a7dda0d0d3f8",
 "log_stream_name" : "lts-topic-kafka"
 }],
 "log_transfer_id" : "ddced522-233a-4181-a5fc-7b458c819afc",
 "log_transfer_info" : {
 "log_create_time" : 1634802241847,
```

```
"log_storage_format": "JSON",
"log_agency_transfer": {
 "agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "agency_domain_name": "paas_apm_z004xxxxx_xx",
 "agency_name": "test20210325",
 "agency_project_id": "2a473356cca5487f8373be891bfxxxxx",
 "be_agency_domain_id": "1d26cc8c86a840e28a4f8d0d078xxxxx",
 "be_agency_project_id": "2a473356cca5487f8373be891bfxxxxx"
},
"log_transfer_detail": {
 "kafka_id": "xxxxxx",
 "kafka_topic": "xxxxxx"
},
"log_transfer_mode": "cycle",
"log_transfer_status": "ENABLE",
"log_transfer_type": "OBS"
}
}]
}
```

**状态码：400**

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid log transfer type"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0010",
 "error_msg": "The system encountered an internal error"
}
```

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 查询转储请求响应成功。                           |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.6.6 注册 DMS kafka 实例

### 功能介绍

该接口用于注册DMS kafka实例。

### 📖 说明

在注册DMS kafka实例前，需要在安全组中，开放入方向规则198.19.128.0/17和9011端口。  
如果DMS kafka实例的子网配置了ACL，则需要在该子网的网络ACL中，开放入方向规则198.19.128.0/17和9011端口。

## 调用方法

请参见[如何调用API](#)。

## URI

POST /v2/{project\_id}/lts/dms/kafka-instance

表 6-357 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-358 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-359 请求 Body 参数

| 参数           | 是否必选 | 参数类型                          | 描述                                            |
|--------------|------|-------------------------------|-----------------------------------------------|
| instance_id  | 是    | String                        | kafka ID<br>最小长度：1<br>最大长度：36                 |
| kafka_name   | 是    | String                        | kafka 名称<br>最小长度：1<br>最大长度：256                |
| connect_info | 是    | <b>connect_info</b><br>object | kafka连接信息。购买kafka时，SASL_SSL是否配置，如果有，则需要填写账号密码 |

表 6-360 connect\_info

| 参数        | 是否必选 | 参数类型   | 描述                       |
|-----------|------|--------|--------------------------|
| user_name | 否    | String | 账号<br>最小长度：1<br>最大长度：256 |
| pwd       | 否    | String | 密码<br>最小长度：1<br>最大长度：256 |

## 响应参数

状态码：201

表 6-361 响应 Body 参数

| 参数          | 参数类型   | 描述       |
|-------------|--------|----------|
| instance_id | String | kafka ID |

状态码：400

表 6-362 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-363 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

注册DMS kafka实例

```
POST https://{endpoint}/v2/{project_id}/lts/dms/kafka-instance
{
 "instance_id" : "3f4a92ff-7f65-489f-a36a-fadbaaxxxx6",
 "kafka_name" : "kafka-wxxxxt",
 "connect_info" : {
 "user_name" : "111",
 "pwd" : "2222"
 }
}
```

## 响应示例

状态码：201

注册DmsKafka请求响应成功。

```
{
 "instance_id" : "3f4a92ff-7f65-489f-a36a-fadbaaxxxx6"
}
```

状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.1719",
 "error_msg" : "kafka user or password wrong"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0010",
 "error_msg" : "The system encountered an internal error"
}
```

**SDK 代码示例**

SDK代码示例如下。

**Java****注册DMS kafka实例**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class RegisterDmsKafkaInstanceSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 RegisterDmsKafkaInstanceRequest request = new RegisterDmsKafkaInstanceRequest();
 RegisterDmsKafkaInstanceRequestBody body = new RegisterDmsKafkaInstanceRequestBody();
 RegisterDmsKafkaInstanceRequestBodyConnectInfo connectInfobody = new
 RegisterDmsKafkaInstanceRequestBodyConnectInfo();
 connectInfobody.withUserName("111")
 .withPwd("2222");
 body.withConnectInfo(connectInfobody);
 body.withKafkaName("kafka-wxxxxt");
 body.withInstanceId("3f4a92ff-7f65-489f-a36a-fadbaaxxxx6");
 request.withBody(body);
 try {
 RegisterDmsKafkaInstanceResponse response = client.registerDmsKafkaInstance(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 }
 }
}
```

```
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 注册DMS kafka实例

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = RegisterDmsKafkaInstanceRequest()
 connectInfoBody = RegisterDmsKafkaInstanceRequestBodyConnectInfo(
 user_name="111",
 pwd="2222"
)
 request.body = RegisterDmsKafkaInstanceRequestBody(
 connect_info=connectInfoBody,
 kafka_name="kafka-wxxxxxt",
 instance_id="3f4a92ff-7f65-489f-a36a-fadbaaxxxx6"
)
 response = client.register_dms_kafka_instance(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 注册DMS kafka实例

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.RegisterDmsKafkaInstanceRequest{}
 userNameConnectInfo := "111"
 pwdConnectInfo := "2222"
 connectInfoBody := &model.RegisterDmsKafkaInstanceRequestBodyConnectInfo{
 UserName: &userNameConnectInfo,
 Pwd: &pwdConnectInfo,
 }
 request.Body = &model.RegisterDmsKafkaInstanceRequestBody{
 ConnectInfo: connectInfoBody,
 KafkaName: "kafka-wxxxxxt",
 InstanceId: "3f4a92ff-7f65-489f-a36a-fadbaaxxxx6",
 }
 response, err := client.RegisterDmsKafkaInstance(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 201 | 注册DmsKafka请求响应成功。                     |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.7 超额采集

### 6.7.1 关闭超额采集开关

#### 功能介绍

该接口用于将超额采集日志功能关闭。

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v2/{project\_id}/collection/disable

表 6-364 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

#### 请求参数

表 6-365 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                                                              |
|--------------|------|--------|-------------------------------------------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>缺省值： <b>None</b><br>最小长度： <b>30</b><br>最大长度： <b>30</b> |

## 响应参数

状态码：403

表 6-366 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

状态码：500

表 6-367 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 请求示例

关闭超额采集开关

```
POST https://{endpoint}/v2/{project_id}/collection/disable
/v2/{project_id}/collection/disable
```

## 响应示例

### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

### 状态码：500

更新超额采集开关状态失败。

```
{
 "error_code": "LTS.0210",
 "error_msg": "Update continue Collection Status error."
}
```

## 状态码

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 200 | 请求响应成功。                                                                                       |
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | 更新超额采集开关状态失败。                                                                                 |

## 错误码

请参见[错误码](#)。

## 6.7.2 打开超额采集开关

### 功能介绍

该接口用于将超额采集日志功能打开。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/collection/enable

表 6-368 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-369 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

## 响应参数

状态码: 403

表 6-370 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值: <ul style="list-style-type: none"><li>LTS.0403</li></ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值: <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

状态码: 500

表 6-371 响应 Body 参数

| 参数         | 参数类型   | 描述                                         |
|------------|--------|--------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• LTS.0403                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• Invalid projectId |

## 请求示例

打开超额采集开关

```
POST https://{endpoint}/v2/{project_id}/collection/enable
/v2/{project_id}/collection/enable
```

## 响应示例

### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

### 状态码：500

更新超额采集开关状态失败。

```
{
 "error_code": "LTS.0210",
 "error_msg": "Update continue Collection Status error."
}
```

## 状态码

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 200 | 请求响应成功。                                                                                       |
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | 更新超额采集开关状态失败。                                                                                 |
| 503 | ServiceUnavailable。被请求的服务无效，服务不可用。                                                            |

## 错误码

请参见[错误码](#)。

## 6.8 云端结构化配置

### 6.8.1 创建结构化配置（推荐）

#### 功能介绍

该接口通过结构化模板创建结构化配置，便于参数提取且简化参数结构，推荐您使用。

单个用户每秒仅能调用1次该接口。

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v3/{project\_id}/lts/struct/template

表 6-372 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

#### 请求参数

表 6-373 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                          |
|--------------|------|--------|-------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30 |

表 6-374 请求 Body 参数

| 参数            | 是否必选 | 参数类型                                        | 描述                                                                                                                                                                                                             |
|---------------|------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_group_id  | 是    | String                                      | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36                                                                                                                                  |
| log_stream_id | 是    | String                                      | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36                                                                                                                                  |
| template_id   | 是    | String                                      | 所用模板id。当使用系统模板时，当前属性可以为空<br>最小长度：0<br>最大长度：36                                                                                                                                                                  |
| template_name | 是    | String                                      | 所用模板名称，会对模板名称及id进行校验<br>最小长度：1<br>最大长度：64                                                                                                                                                                      |
| template_type | 是    | String                                      | 所用模板类型，分为built_in及custom两种类型，对应系统模板和自定义模板，详细系统模板类型参考：“ <a href="#">用户指南-日志搜索与分析（默认推荐）&gt; 云端结构化解析&gt; 结构化模板</a> ”章节。<br>枚举值：<br><ul style="list-style-type: none"> <li>● built_in</li> <li>● custom</li> </ul> |
| demo_fields   | 否    | Array of <a href="#">FieldModel</a> objects | 示例字段数组，填写与模板中is_analysis参数状态不同的字段。                                                                                                                                                                             |

| 参数             | 是否必选 | 参数类型                               | 描述                                                                                                                                           |
|----------------|------|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| tag_fields     | 否    | Array of <b>FieldModel</b> objects | Tag字段数组，填写与模板中 is_analysis参数状态不同的字段。                                                                                                         |
| quick_analysis | 否    | Boolean                            | 是否开启demo_fields和 tag_fields快速分析,为true时,所有的demo_fields和tag_fields全部字段均打开快速分析;不填或者为false,以模板中的 demo_fields和tag_fields中的 is_analysis决定是否开启快速分析。 |

表 6-375 FieldModel

| 参数          | 是否必选 | 参数类型    | 描述                        |
|-------------|------|---------|---------------------------|
| field_name  | 是    | String  | 字段名称<br>最小长度：1<br>最大长度：64 |
| is_analysis | 否    | Boolean | 是否开启快速分析。                 |

## 响应参数

状态码：400

表 6-376 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                   |
|------------|--------|--------------------------------------------------------------------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：8                                                              |
| error_msg  | String | 调用失败响应信息描述<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

状态码：500

表 6-377 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

- 创建ELB系统模板

POST https://{endpoint}/v3/{project\_id}/lts/struct/template

```
{
 "log_group_id": "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
 "log_stream_id": "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 "demo_fields": [{
 "field_name": "msec",
 "is_analysis": false
 }, {
 "field_name": "access_log_topic_id",
 "is_analysis": false
 }, {
 "field_name": "time_iso8601",
 "is_analysis": false
 }, {
 "field_name": "log_ver",
 "is_analysis": true
 }, {
 "field_name": "remote_addr",
 "is_analysis": true
 }, {
 "field_name": "remote_port",
 "is_analysis": false
 }, {
 "field_name": "status",
 "is_analysis": false
 }, {
 "field_name": "request_method",
 "is_analysis": false
 }, {
 "field_name": "scheme",
 "is_analysis": true
 }, {
 "field_name": "host",
 "is_analysis": true
 }, {
 "field_name": "router_request_uri",
 "is_analysis": true
 }, {
 "field_name": "server_protocol",
 "is_analysis": true
 }, {
 "field_name": "request_length",
 "is_analysis": true
 }, {
 "field_name": "bytes_sent",
 "is_analysis": false
 }, {
 "field_name": "body_bytes_sent",
 "is_analysis": false
 }, {
 }, {
```

```
"field_name": "request_time",
"is_analysis": false
}, {
 "field_name": "upstream_status",
 "is_analysis": false
}, {
 "field_name": "upstream_connect_time",
 "is_analysis": false
}, {
 "field_name": "upstream_header_time",
 "is_analysis": false
}, {
 "field_name": "upstream_response_time",
 "is_analysis": false
}, {
 "field_name": "upstream_addr",
 "is_analysis": false
}, {
 "field_name": "http_user_agent",
 "is_analysis": false
}, {
 "field_name": "http_referer",
 "is_analysis": false
}, {
 "field_name": "http_x_forwarded_for",
 "is_analysis": false
}, {
 "field_name": "lb_name",
 "is_analysis": false
}, {
 "field_name": "listener_name",
 "is_analysis": false
}, {
 "field_name": "listener_id",
 "is_analysis": false
}, {
 "field_name": "pool_name",
 "is_analysis": false
}, {
 "field_name": "member_name",
 "is_analysis": false
}, {
 "field_name": "tenant_id",
 "is_analysis": false
}, {
 "field_name": "eip_address",
 "is_analysis": false
}, {
 "field_name": "eip_port",
 "is_analysis": false
}, {
 "field_name": "upstream_addr_priv",
 "is_analysis": false
}, {
 "field_name": "certificate_id",
 "is_analysis": false
}, {
 "field_name": "ssl_protocol",
 "is_analysis": false
}, {
 "field_name": "ssl_cipher",
 "is_analysis": false
}, {
 "field_name": "sni_domain_name",
 "is_analysis": false
}, {
 "field_name": "tcpinfo_rtt",
 "is_analysis": false
}],
```

```
"tag_fields" : [{
 "field_name" : "hostIP",
 "is_analysis" : true
},
"template_type" : "built_in",
"template_name" : "ELB",
"template_id" : "",
"quick_analysis" : false
}
```

- 创建VPC系统模板

POST https://{endpoint}/v3/{project\_id}/lts/struct/template

```
{
 "log_group_id" : "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
 "log_stream_id" : "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 "demo_fields" : [{
 "field_name" : "version",
 "is_analysis" : false
 }, {
 "field_name" : "project_id",
 "is_analysis" : true
 }, {
 "field_name" : "interface_id",
 "is_analysis" : false
 }, {
 "field_name" : "srcaddr",
 "is_analysis" : true
 }, {
 "field_name" : "dstaddr",
 "is_analysis" : true
 }, {
 "field_name" : "srcport",
 "is_analysis" : false
 }, {
 "field_name" : "dstport",
 "is_analysis" : false
 }, {
 "field_name" : "protocol",
 "is_analysis" : false
 }, {
 "field_name" : "packets",
 "is_analysis" : false
 }, {
 "field_name" : "bytes",
 "is_analysis" : false
 }, {
 "field_name" : "start",
 "is_analysis" : false
 }, {
 "field_name" : "end",
 "is_analysis" : false
 }, {
 "field_name" : "action",
 "is_analysis" : true
 }, {
 "field_name" : "log_status",
 "is_analysis" : true
 }],
 "tag_fields" : [{
 "field_name" : "hostIP",
 "is_analysis" : true
 }],
 "template_type" : "built_in",
 "template_name" : "VPC",
 "template_id" : "",
 "quick_analysis" : false
}
```

## 响应示例

### 状态码：201

请求响应成功。

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.2014",
 "error_msg" : "template_id is invalid!"
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2014",
 "error_msg" : "Failed to create struct config."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 创建ELB系统模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateStructConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
```

```
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
CreateStructConfigRequest request = new CreateStructConfigRequest();
StructConfig body = new StructConfig();
List<FieldModel> listbodyTagFields = new ArrayList<>();
listbodyTagFields.add(
 new FieldModel()
 .withFieldName("hostIP")
 .withIsAnalysis(true)
);
List<FieldModel> listbodyDemoFields = new ArrayList<>();
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("msec")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("access_log_topic_id")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("time_iso8601")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("log_ver")
 .withIsAnalysis(true)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("remote_addr")
 .withIsAnalysis(true)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("remote_port")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("status")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_method")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("scheme")
 .withIsAnalysis(true)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("host")
 .withIsAnalysis(true)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("router_request_uri")
 .withIsAnalysis(true)
);
listbodyDemoFields.add(
 new FieldModel()
```

```
 .withFieldName("server_protocol")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_length")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("bytes_sent")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("body_bytes_sent")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_time")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_status")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_connect_time")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_header_time")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_response_time")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_addr")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_user_agent")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_referer")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_x_forwarded_for")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("lb_name")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
```

```
 new FieldModel()
 .withFieldName("listener_name")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("listener_id")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("pool_name")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("member_name")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("tenant_id")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("eip_address")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("eip_port")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_addr_priv")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("certificate_id")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("ssl_protocol")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("ssl_cipher")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("sni_domain_name")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("tcpinfo_rtt")
 .withIsAnalysis(false)
);
);
 body.withQuickAnalysis(false);
 body.withTagFields(listbodyTagFields);
 body.withDemoFields(listbodyDemoFields);
 body.withTemplateType(StructConfig.TemplateTypeEnum.fromValue("built_in"));
 body.withTemplateName("ELB");
```

```
body.withTemplateId("");
body.withLogStreamId("b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx");
body.withLogGroupId("17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx");
request.withBody(body);
try {
 CreateStructConfigResponse response = client.createStructConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- 创建VPC系统模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateStructConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateStructConfigRequest request = new CreateStructConfigRequest();
 StructConfig body = new StructConfig();
 List<FieldModel> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new FieldModel()
 .withFieldName("hostIP")
 .withIsAnalysis(true)
);
 List<FieldModel> listbodyDemoFields = new ArrayList<>();
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("version")
);
 }
}
```

```
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("project_id")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("interface_id")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("srcaddr")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("dstaddr")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("srcport")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("dstport")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("protocol")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("packets")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("bytes")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("start")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("end")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("action")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("log_status")
 .withIsAnalysis(true)
);
 body.withQuickAnalysis(false);
 body.withTagFields(listbodyTagFields);
```

```
body.withDemoFields(listbodyDemoFields);
body.withTemplateType(StructConfig.TemplateTypeEnum.fromValue("built_in"));
body.withTemplateName("VPC");
body.withTemplateId("");
body.withLogStreamId("b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx");
body.withLogGroupId("17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx");
request.withBody(body);
try {
 CreateStructConfigResponse response = client.createStructConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 创建ELB系统模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateStructConfigRequest()
 listTagFieldsbody = [
 FieldModel(
 field_name="hostIP",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 FieldModel(
 field_name="msec",
 is_analysis=False
),
 FieldModel(
 field_name="access_log_topic_id",
 is_analysis=False
)
]
```

```
),
FieldModel(
 field_name="time_iso8601",
 is_analysis=False
),
FieldModel(
 field_name="log_ver",
 is_analysis=True
),
FieldModel(
 field_name="remote_addr",
 is_analysis=True
),
FieldModel(
 field_name="remote_port",
 is_analysis=False
),
FieldModel(
 field_name="status",
 is_analysis=False
),
FieldModel(
 field_name="request_method",
 is_analysis=False
),
FieldModel(
 field_name="scheme",
 is_analysis=True
),
FieldModel(
 field_name="host",
 is_analysis=True
),
FieldModel(
 field_name="router_request_uri",
 is_analysis=True
),
FieldModel(
 field_name="server_protocol",
 is_analysis=True
),
FieldModel(
 field_name="request_length",
 is_analysis=True
),
FieldModel(
 field_name="bytes_sent",
 is_analysis=False
),
FieldModel(
 field_name="body_bytes_sent",
 is_analysis=False
),
FieldModel(
 field_name="request_time",
 is_analysis=False
),
FieldModel(
 field_name="upstream_status",
 is_analysis=False
),
FieldModel(
 field_name="upstream_connect_time",
 is_analysis=False
),
FieldModel(
 field_name="upstream_header_time",
 is_analysis=False
),
),
```

```
FieldModel(
 field_name="upstream_response_time",
 is_analysis=False
),
FieldModel(
 field_name="upstream_addr",
 is_analysis=False
),
FieldModel(
 field_name="http_user_agent",
 is_analysis=False
),
FieldModel(
 field_name="http_referer",
 is_analysis=False
),
FieldModel(
 field_name="http_x_forwarded_for",
 is_analysis=False
),
FieldModel(
 field_name="lb_name",
 is_analysis=False
),
FieldModel(
 field_name="listener_name",
 is_analysis=False
),
FieldModel(
 field_name="listener_id",
 is_analysis=False
),
FieldModel(
 field_name="pool_name",
 is_analysis=False
),
FieldModel(
 field_name="member_name",
 is_analysis=False
),
FieldModel(
 field_name="tenant_id",
 is_analysis=False
),
FieldModel(
 field_name="eip_address",
 is_analysis=False
),
FieldModel(
 field_name="eip_port",
 is_analysis=False
),
FieldModel(
 field_name="upstream_addr_priv",
 is_analysis=False
),
FieldModel(
 field_name="certificate_id",
 is_analysis=False
),
FieldModel(
 field_name="ssl_protocol",
 is_analysis=False
),
FieldModel(
 field_name="ssl_cipher",
 is_analysis=False
),
FieldModel(

```

```
 field_name="sni_domain_name",
 is_analysis=False
),
 FieldModel(
 field_name="tcpinfo_rtt",
 is_analysis=False
)
]
request.body = StructConfig(
 quick_analysis=False,
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody,
 template_type="built_in",
 template_name="ELB",
 template_id="",
 log_stream_id="b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 log_group_id="17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx"
)
response = client.create_struct_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 创建VPC系统模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateStructConfigRequest()
 listTagFieldsbody = [
 FieldModel(
 field_name="hostIP",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 FieldModel(
 field_name="version",
 is_analysis=False
),
 FieldModel(
 field_name="project_id",
 is_analysis=True
),
],
```

```
FieldModel(
 field_name="interface_id",
 is_analysis=False
),
FieldModel(
 field_name="srcaddr",
 is_analysis=True
),
FieldModel(
 field_name="dstaddr",
 is_analysis=True
),
FieldModel(
 field_name="srcport",
 is_analysis=False
),
FieldModel(
 field_name="dstport",
 is_analysis=False
),
FieldModel(
 field_name="protocol",
 is_analysis=False
),
FieldModel(
 field_name="packets",
 is_analysis=False
),
FieldModel(
 field_name="bytes",
 is_analysis=False
),
FieldModel(
 field_name="start",
 is_analysis=False
),
FieldModel(
 field_name="end",
 is_analysis=False
),
FieldModel(
 field_name="action",
 is_analysis=True
),
FieldModel(
 field_name="log_status",
 is_analysis=True
)
]
request.body = StructConfig(
 quick_analysis=False,
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody,
 template_type="built_in",
 template_name="VPC",
 template_id="",
 log_stream_id="b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 log_group_id="17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx"
)
response = client.create_struct_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 创建ELB系统模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateStructConfigRequest{}
 isAnalysisTagFields:= true
 var listTagFieldsbody = []model.FieldModel{
 {
 FieldName: "hostIP",
 IsAnalysis: &isAnalysisTagFields,
 },
 }
 isAnalysisDemoFields:= false
 isAnalysisDemoFields1:= false
 isAnalysisDemoFields2:= false
 isAnalysisDemoFields3:= true
 isAnalysisDemoFields4:= true
 isAnalysisDemoFields5:= false
 isAnalysisDemoFields6:= false
 isAnalysisDemoFields7:= false
 isAnalysisDemoFields8:= true
 isAnalysisDemoFields9:= true
 isAnalysisDemoFields10:= true
 isAnalysisDemoFields11:= true
 isAnalysisDemoFields12:= true
 isAnalysisDemoFields13:= false
 isAnalysisDemoFields14:= false
 isAnalysisDemoFields15:= false
 isAnalysisDemoFields16:= false
 isAnalysisDemoFields17:= false
 isAnalysisDemoFields18:= false
 isAnalysisDemoFields19:= false
 isAnalysisDemoFields20:= false
 isAnalysisDemoFields21:= false
 isAnalysisDemoFields22:= false
 isAnalysisDemoFields23:= false
 isAnalysisDemoFields24:= false
}
```

```
isAnalysisDemoFields25:= false
isAnalysisDemoFields26:= false
isAnalysisDemoFields27:= false
isAnalysisDemoFields28:= false
isAnalysisDemoFields29:= false
isAnalysisDemoFields30:= false
isAnalysisDemoFields31:= false
isAnalysisDemoFields32:= false
isAnalysisDemoFields33:= false
isAnalysisDemoFields34:= false
isAnalysisDemoFields35:= false
isAnalysisDemoFields36:= false
isAnalysisDemoFields37:= false
var listDemoFieldsbody = []model.FieldModel{
 {
 FieldName: "msec",
 IsAnalysis: &isAnalysisDemoFields,
 },
 {
 FieldName: "access_log_topic_id",
 IsAnalysis: &isAnalysisDemoFields1,
 },
 {
 FieldName: "time_iso8601",
 IsAnalysis: &isAnalysisDemoFields2,
 },
 {
 FieldName: "log_ver",
 IsAnalysis: &isAnalysisDemoFields3,
 },
 {
 FieldName: "remote_addr",
 IsAnalysis: &isAnalysisDemoFields4,
 },
 {
 FieldName: "remote_port",
 IsAnalysis: &isAnalysisDemoFields5,
 },
 {
 FieldName: "status",
 IsAnalysis: &isAnalysisDemoFields6,
 },
 {
 FieldName: "request_method",
 IsAnalysis: &isAnalysisDemoFields7,
 },
 {
 FieldName: "scheme",
 IsAnalysis: &isAnalysisDemoFields8,
 },
 {
 FieldName: "host",
 IsAnalysis: &isAnalysisDemoFields9,
 },
 {
 FieldName: "router_request_uri",
 IsAnalysis: &isAnalysisDemoFields10,
 },
 {
 FieldName: "server_protocol",
 IsAnalysis: &isAnalysisDemoFields11,
 },
 {
 FieldName: "request_length",
 IsAnalysis: &isAnalysisDemoFields12,
 },
 {
 FieldName: "bytes_sent",
 IsAnalysis: &isAnalysisDemoFields13,
```

```
},
{
 FieldName: "body_bytes_sent",
 IsAnalysis: &isAnalysisDemoFields14,
},
{
 FieldName: "request_time",
 IsAnalysis: &isAnalysisDemoFields15,
},
{
 FieldName: "upstream_status",
 IsAnalysis: &isAnalysisDemoFields16,
},
{
 FieldName: "upstream_connect_time",
 IsAnalysis: &isAnalysisDemoFields17,
},
{
 FieldName: "upstream_header_time",
 IsAnalysis: &isAnalysisDemoFields18,
},
{
 FieldName: "upstream_response_time",
 IsAnalysis: &isAnalysisDemoFields19,
},
{
 FieldName: "upstream_addr",
 IsAnalysis: &isAnalysisDemoFields20,
},
{
 FieldName: "http_user_agent",
 IsAnalysis: &isAnalysisDemoFields21,
},
{
 FieldName: "http_referer",
 IsAnalysis: &isAnalysisDemoFields22,
},
{
 FieldName: "http_x_forwarded_for",
 IsAnalysis: &isAnalysisDemoFields23,
},
{
 FieldName: "lb_name",
 IsAnalysis: &isAnalysisDemoFields24,
},
{
 FieldName: "listener_name",
 IsAnalysis: &isAnalysisDemoFields25,
},
{
 FieldName: "listener_id",
 IsAnalysis: &isAnalysisDemoFields26,
},
{
 FieldName: "pool_name",
 IsAnalysis: &isAnalysisDemoFields27,
},
{
 FieldName: "member_name",
 IsAnalysis: &isAnalysisDemoFields28,
},
{
 FieldName: "tenant_id",
 IsAnalysis: &isAnalysisDemoFields29,
},
{
 FieldName: "eip_address",
 IsAnalysis: &isAnalysisDemoFields30,
},
},
```

```
{
 {
 FileName: "eip_port",
 IsAnalysis: &isAnalysisDemoFields31,
 },
 {
 FileName: "upstream_addr_priv",
 IsAnalysis: &isAnalysisDemoFields32,
 },
 {
 FileName: "certificate_id",
 IsAnalysis: &isAnalysisDemoFields33,
 },
 {
 FileName: "ssl_protocol",
 IsAnalysis: &isAnalysisDemoFields34,
 },
 {
 FileName: "ssl_cipher",
 IsAnalysis: &isAnalysisDemoFields35,
 },
 {
 FileName: "sni_domain_name",
 IsAnalysis: &isAnalysisDemoFields36,
 },
 {
 FileName: "tcpinfo_rtt",
 IsAnalysis: &isAnalysisDemoFields37,
 },
}
quickAnalysisStructConfig:= false
request.Body = &model.StructConfig{
 QuickAnalysis: &quickAnalysisStructConfig,
 TagFields: &listTagFieldsbody,
 DemoFields: &listDemoFieldsbody,
 TemplateType: model.GetStructConfigTemplateTypeEnum().BUILT_IN,
 TemplateName: "ELB",
 TemplateId: "",
 LogStreamId: "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 LogGroupId: "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
}
response, err := client.CreateStructConfig(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

- 创建VPC系统模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.CreateStructConfigRequest{
 isAnalysisTagFields:= true
 var listTagFieldsbody = []model.FieldModel{
 {
 FieldName: "hostIP",
 IsAnalysis: &isAnalysisTagFields,
 },
 }
 isAnalysisDemoFields:= false
 isAnalysisDemoFields1:= true
 isAnalysisDemoFields2:= false
 isAnalysisDemoFields3:= true
 isAnalysisDemoFields4:= true
 isAnalysisDemoFields5:= false
 isAnalysisDemoFields6:= false
 isAnalysisDemoFields7:= false
 isAnalysisDemoFields8:= false
 isAnalysisDemoFields9:= false
 isAnalysisDemoFields10:= false
 isAnalysisDemoFields11:= false
 isAnalysisDemoFields12:= true
 isAnalysisDemoFields13:= true
 var listDemoFieldsbody = []model.FieldModel{
 {
 FieldName: "version",
 IsAnalysis: &isAnalysisDemoFields,
 },
 {
 FieldName: "project_id",
 IsAnalysis: &isAnalysisDemoFields1,
 },
 {
 FieldName: "interface_id",
 IsAnalysis: &isAnalysisDemoFields2,
 },
 {
 FieldName: "srcaddr",
 IsAnalysis: &isAnalysisDemoFields3,
 },
 {
 FieldName: "dstaddr",
 IsAnalysis: &isAnalysisDemoFields4,
 },
 {
 FieldName: "srcport",
 IsAnalysis: &isAnalysisDemoFields5,
 },
 {
 FieldName: "dstport",
 IsAnalysis: &isAnalysisDemoFields6,
 },
 {
 FieldName: "protocol",
 IsAnalysis: &isAnalysisDemoFields7,
 },
 {
 FieldName: "packets",
```

```
 IsAnalysis: &isAnalysisDemoFields8,
 },
 {
 FileName: "bytes",
 IsAnalysis: &isAnalysisDemoFields9,
 },
 {
 FileName: "start",
 IsAnalysis: &isAnalysisDemoFields10,
 },
 {
 FileName: "end",
 IsAnalysis: &isAnalysisDemoFields11,
 },
 {
 FileName: "action",
 IsAnalysis: &isAnalysisDemoFields12,
 },
 {
 FileName: "log_status",
 IsAnalysis: &isAnalysisDemoFields13,
 },
}
quickAnalysisStructConfig:= false
request.Body = &model.StructConfig{
 QuickAnalysis: &quickAnalysisStructConfig,
 TagFields: &listTagFieldsbody,
 DemoFields: &listDemoFieldsbody,
 TemplateType: model.GetStructConfigTemplateTypeEnum().BUILT_IN,
 TemplateName: "VPC",
 TemplateId: "",
 LogStreamId: "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 LogGroupId: "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
}
response, err := client.CreateStructConfig(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 201 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.8.2 修改结构化配置（推荐）

### 功能介绍

该接口通过结构化模板修改结构化配置

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v3/{project\_id}/lts/struct/template

表 6-378 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-379 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-380 请求 Body 参数

| 参数            | 是否必选 | 参数类型                                        | 描述                                                                                                                                                                               |
|---------------|------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_group_id  | 是    | String                                      | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36                                                                                                    |
| log_stream_id | 是    | String                                      | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36                                                                                                    |
| template_id   | 是    | String                                      | 所用模板id。当使用系统模板时，当前属性可以为空<br>最小长度：0<br>最大长度：36                                                                                                                                    |
| template_name | 是    | String                                      | 所用模板名称，会对模板名称及id进行校验<br>最小长度：1<br>最大长度：64                                                                                                                                        |
| template_type | 是    | String                                      | 所用模板类型，分为built_in及custom两种类型，对应系统模板和自定义模板，详细系统模板类型参考：“用户指南-日志搜索与分析（默认推荐）> 云端结构化解析> 结构化模板”章节。<br>枚举值： <ul style="list-style-type: none"><li>• built_in</li><li>• custom</li></ul> |
| demo_fields   | 否    | Array of <a href="#">FieldModel</a> objects | 示例字段数组，填写与模板中is_analysis参数状态不同的字段。                                                                                                                                               |
| tag_fields    | 否    | Array of <a href="#">FieldModel</a> objects | Tag字段数组，填写与模板中is_analysis参数状态不同的字段。                                                                                                                                              |

| 参数             | 是否必选 | 参数类型    | 描述                                                                                                                                        |
|----------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------|
| quick_analysis | 否    | Boolean | 是否开启demo_fields和tag_fields快速分析,为true时,所有的demo_fields和tag_fields全部字段均打开快速分析;不填或者为false,以模板中的demo_fields和tag_fields中的is_analysis决定是否开启快速分析。 |

表 6-381 FieldModel

| 参数          | 是否必选 | 参数类型    | 描述                          |
|-------------|------|---------|-----------------------------|
| field_name  | 是    | String  | 字段名称<br>最小长度: 1<br>最大长度: 64 |
| is_analysis | 否    | Boolean | 是否开启快速分析。                   |

## 响应参数

状态码: 201

表 6-382 响应 Body 参数

| 参数 | 参数类型   | 描述 |
|----|--------|----|
| -  | String |    |

状态码: 400

表 6-383 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码。<br>最小长度: 8<br>最大长度: 8 |
| error_msg  | String | 调用失败响应信息描述。                |

状态码: 500

表 6-384 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

- 修改ELB系统模板

PUT https://{endpoint}/v3/{project\_id}/lts/struct/template

```
{
 "log_group_id": "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
 "log_stream_id": "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 "demo_fields": [{
 "field_name": "msec",
 "is_analysis": false
 }, {
 "field_name": "access_log_topic_id",
 "is_analysis": false
 }, {
 "field_name": "time_iso8601",
 "is_analysis": false
 }, {
 "field_name": "log_ver",
 "is_analysis": true
 }, {
 "field_name": "remote_addr",
 "is_analysis": true
 }, {
 "field_name": "remote_port",
 "is_analysis": false
 }, {
 "field_name": "status",
 "is_analysis": false
 }, {
 "field_name": "request_method",
 "is_analysis": false
 }, {
 "field_name": "scheme",
 "is_analysis": true
 }, {
 "field_name": "host",
 "is_analysis": true
 }, {
 "field_name": "router_request_uri",
 "is_analysis": true
 }, {
 "field_name": "server_protocol",
 "is_analysis": true
 }, {
 "field_name": "request_length",
 "is_analysis": true
 }, {
 "field_name": "bytes_sent",
 "is_analysis": false
 }, {
 "field_name": "body_bytes_sent",
 "is_analysis": false
 }, {
 }
```

```
"field_name": "request_time",
"is_analysis": false
}, {
 "field_name": "upstream_status",
 "is_analysis": false
}, {
 "field_name": "upstream_connect_time",
 "is_analysis": false
}, {
 "field_name": "upstream_header_time",
 "is_analysis": false
}, {
 "field_name": "upstream_response_time",
 "is_analysis": false
}, {
 "field_name": "upstream_addr",
 "is_analysis": false
}, {
 "field_name": "http_user_agent",
 "is_analysis": false
}, {
 "field_name": "http_referer",
 "is_analysis": false
}, {
 "field_name": "http_x_forwarded_for",
 "is_analysis": false
}, {
 "field_name": "lb_name",
 "is_analysis": false
}, {
 "field_name": "listener_name",
 "is_analysis": false
}, {
 "field_name": "listener_id",
 "is_analysis": false
}, {
 "field_name": "pool_name",
 "is_analysis": false
}, {
 "field_name": "member_name",
 "is_analysis": false
}, {
 "field_name": "tenant_id",
 "is_analysis": false
}, {
 "field_name": "eip_address",
 "is_analysis": false
}, {
 "field_name": "eip_port",
 "is_analysis": false
}, {
 "field_name": "upstream_addr_priv",
 "is_analysis": false
}, {
 "field_name": "certificate_id",
 "is_analysis": false
}, {
 "field_name": "ssl_protocol",
 "is_analysis": false
}, {
 "field_name": "ssl_cipher",
 "is_analysis": false
}, {
 "field_name": "sni_domain_name",
 "is_analysis": false
}, {
 "field_name": "tcpinfo_rtt",
 "is_analysis": false
}],
```

```
"tag_fields" : [{
 "field_name" : "hostIP",
 "is_analysis" : true
}],
"template_type" : "built_in",
"template_name" : "ELB",
"template_id" : "",
"quick_analysis" : false
}
```

- 修改VPC系统模板

PUT https://{endpoint}/v3/{project\_id}/lts/struct/template

```
{
 "log_group_id" : "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
 "log_stream_id" : "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 "demo_fields" : [{
 "field_name" : "version"
 }, {
 "field_name" : "project_id"
 }, {
 "field_name" : "interface_id"
 }, {
 "field_name" : "srcaddr"
 }, {
 "field_name" : "dstaddr"
 }, {
 "field_name" : "srcport"
 }, {
 "field_name" : "dstport"
 }, {
 "field_name" : "protocol"
 }, {
 "field_name" : "packets"
 }, {
 "field_name" : "bytes"
 }, {
 "field_name" : "start"
 }, {
 "field_name" : "end"
 }, {
 "field_name" : "action"
 }, {
 "field_name" : "log_status"
 }],
 "tag_fields" : [{
 "field_name" : "hostIP",
 "is_analysis" : true
 }],
 "template_type" : "built_in",
 "template_name" : "VPC",
 "template_id" : "",
 "quick_analysis" : false
}
```

## 响应示例

### 状态码：201

请求响应成功。

```
""
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.2014",
```

```
"error_msg" : "Failed to create struct config."
"preciseEndTime" : 0,
"preciseStartTime" : 0
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2016",
 "error_msg" : "Failed to update struct config"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 修改ELB系统模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateStructConfigSolution {
 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateStructConfigRequest request = new UpdateStructConfigRequest();
 StructConfig body = new StructConfig();
 List<FieldModel> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new FieldModel()
 .withFieldName("hostIP")
 .withIsAnalysis(true)
);
 List<FieldModel> listbodyDemoFields = new ArrayList<>();
 listbodyDemoFields.add(

```

```
 new FieldModel()
 .withFieldName("msec")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("access_log_topic_id")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("time_iso8601")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("log_ver")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("remote_addr")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("remote_port")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("status")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_method")
 .withIsAnalysis(false)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("scheme")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("host")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("router_request_uri")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("server_protocol")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_length")
 .withIsAnalysis(true)
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("bytes_sent")
 .withIsAnalysis(false)
);
);
```

```
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("body_bytes_sent")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("request_time")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_status")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_connect_time")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_header_time")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_response_time")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_addr")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_user_agent")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_referer")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("http_x_forwarded_for")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("lb_name")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("listener_name")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("listener_id")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("pool_name")
 .withIsAnalysis(false)
);
```

```
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("member_name")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("tenant_id")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("eip_address")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("eip_port")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("upstream_addr_priv")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("certificate_id")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("ssl_protocol")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("ssl_cipher")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("sni_domain_name")
 .withIsAnalysis(false)
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("tcpinfo_rtt")
 .withIsAnalysis(false)
);
body.withQuickAnalysis(false);
body.withTagFields(listbodyTagFields);
body.withDemoFields(listbodyDemoFields);
body.withTemplateType(StructConfig.TemplateTypeEnum.fromValue("built_in"));
body.withTemplateName("ELB");
body.withTemplateId("");
body.withLogStreamId("b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx");
body.withLogGroupId("17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx");
request.withBody(body);
try {
 UpdateStructConfigResponse response = client.updateStructConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
}
```

```
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

- 修改VPC系统模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateStructConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateStructConfigRequest request = new UpdateStructConfigRequest();
 StructConfig body = new StructConfig();
 List<FieldModel> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new FieldModel()
 .withFieldName("hostIP")
 .withIsAnalysis(true)
);
 List<FieldModel> listbodyDemoFields = new ArrayList<>();
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("version")
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("project_id")
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("interface_id")
);
 listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("srcaddr")
);
 }
}
```

```
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("dstaddr")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("srcport")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("dstport")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("protocol")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("packets")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("bytes")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("start")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("end")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("action")
);
listbodyDemoFields.add(
 new FieldModel()
 .withFieldName("log_status")
);
body.withQuickAnalysis(false);
body.withTagFields(listbodyTagFields);
body.withDemoFields(listbodyDemoFields);
body.withTemplateType(StructConfig.TemplateTypeEnum.fromValue("built_in"));
body.withTemplateName("VPC");
body.withTemplateId("");
body.withLogStreamId("b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx");
body.withLogGroupId("17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx");
request.withBody(body);
try {
 UpdateStructConfigResponse response = client.updateStructConfig(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 修改ELB系统模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateStructConfigRequest()
 listTagFieldsbody = [
 FieldModel(
 field_name="hostIP",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 FieldModel(
 field_name="msec",
 is_analysis=False
),
 FieldModel(
 field_name="access_log_topic_id",
 is_analysis=False
),
 FieldModel(
 field_name="time_iso8601",
 is_analysis=False
),
 FieldModel(
 field_name="log_ver",
 is_analysis=True
),
 FieldModel(
 field_name="remote_addr",
 is_analysis=True
),
 FieldModel(
 field_name="remote_port",
 is_analysis=False
),
 FieldModel(
 field_name="status",
 is_analysis=False
),
 FieldModel(
 field_name="request_method",
 is_analysis=False
),
 FieldModel(
 field_name="scheme",
```

```
 is_analysis=True
),
 FieldModel(
 field_name="host",
 is_analysis=True
),
 FieldModel(
 field_name="router_request_uri",
 is_analysis=True
),
 FieldModel(
 field_name="server_protocol",
 is_analysis=True
),
 FieldModel(
 field_name="request_length",
 is_analysis=True
),
 FieldModel(
 field_name="bytes_sent",
 is_analysis=False
),
 FieldModel(
 field_name="body_bytes_sent",
 is_analysis=False
),
 FieldModel(
 field_name="request_time",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_status",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_connect_time",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_header_time",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_response_time",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_addr",
 is_analysis=False
),
 FieldModel(
 field_name="http_user_agent",
 is_analysis=False
),
 FieldModel(
 field_name="http_referer",
 is_analysis=False
),
 FieldModel(
 field_name="http_x_forwarded_for",
 is_analysis=False
),
 FieldModel(
 field_name="lb_name",
 is_analysis=False
),
 FieldModel(
 field_name="listener_name",
 is_analysis=False
)
```

```
),
 FieldModel(
 field_name="listener_id",
 is_analysis=False
),
 FieldModel(
 field_name="pool_name",
 is_analysis=False
),
 FieldModel(
 field_name="member_name",
 is_analysis=False
),
 FieldModel(
 field_name="tenant_id",
 is_analysis=False
),
 FieldModel(
 field_name="eip_address",
 is_analysis=False
),
 FieldModel(
 field_name="eip_port",
 is_analysis=False
),
 FieldModel(
 field_name="upstream_addr_priv",
 is_analysis=False
),
 FieldModel(
 field_name="certificate_id",
 is_analysis=False
),
 FieldModel(
 field_name="ssl_protocol",
 is_analysis=False
),
 FieldModel(
 field_name="ssl_cipher",
 is_analysis=False
),
 FieldModel(
 field_name="sni_domain_name",
 is_analysis=False
),
 FieldModel(
 field_name="tcpinfo_rtt",
 is_analysis=False
)
]
request.body = StructConfig(
 quick_analysis=False,
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody,
 template_type="built_in",
 template_name="ELB",
 template_id="",
 log_stream_id="b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 log_group_id="17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx"
)
response = client.update_struct_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- 修改VPC系统模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateStructConfigRequest()
 listTagFieldsbody = [
 FieldModel(
 field_name="hostIP",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 FieldModel(
 field_name="version"
),
 FieldModel(
 field_name="project_id"
),
 FieldModel(
 field_name="interface_id"
),
 FieldModel(
 field_name="srcaddr"
),
 FieldModel(
 field_name="dstaddr"
),
 FieldModel(
 field_name="srcport"
),
 FieldModel(
 field_name="dstport"
),
 FieldModel(
 field_name="protocol"
),
 FieldModel(
 field_name="packets"
),
 FieldModel(
 field_name="bytes"
),
 FieldModel(
 field_name="start"
),
 FieldModel(
```

```
 field_name="end"
),
 FieldModel(
 field_name="action"
),
 FieldModel(
 field_name="log_status"
)
]
request.body = StructConfig(
 quick_analysis=False,
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody,
 template_type="built_in",
 template_name="VPC",
 template_id="",
 log_stream_id="b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 log_group_id="17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx"
)
response = client.update_struct_config(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 修改ELB系统模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateStructConfigRequest{}
 isAnalysisTagFields:= true
 var listTagFieldsbody = []model.FieldModel{
 {
 FieldName: "hostIP",
 IsAnalysis: &isAnalysisTagFields,
 }
 }
}
```

```
 },
 }
 isAnalysisDemoFields:= false
 isAnalysisDemoFields1:= false
 isAnalysisDemoFields2:= false
 isAnalysisDemoFields3:= true
 isAnalysisDemoFields4:= true
 isAnalysisDemoFields5:= false
 isAnalysisDemoFields6:= false
 isAnalysisDemoFields7:= false
 isAnalysisDemoFields8:= true
 isAnalysisDemoFields9:= true
 isAnalysisDemoFields10:= true
 isAnalysisDemoFields11:= true
 isAnalysisDemoFields12:= true
 isAnalysisDemoFields13:= false
 isAnalysisDemoFields14:= false
 isAnalysisDemoFields15:= false
 isAnalysisDemoFields16:= false
 isAnalysisDemoFields17:= false
 isAnalysisDemoFields18:= false
 isAnalysisDemoFields19:= false
 isAnalysisDemoFields20:= false
 isAnalysisDemoFields21:= false
 isAnalysisDemoFields22:= false
 isAnalysisDemoFields23:= false
 isAnalysisDemoFields24:= false
 isAnalysisDemoFields25:= false
 isAnalysisDemoFields26:= false
 isAnalysisDemoFields27:= false
 isAnalysisDemoFields28:= false
 isAnalysisDemoFields29:= false
 isAnalysisDemoFields30:= false
 isAnalysisDemoFields31:= false
 isAnalysisDemoFields32:= false
 isAnalysisDemoFields33:= false
 isAnalysisDemoFields34:= false
 isAnalysisDemoFields35:= false
 isAnalysisDemoFields36:= false
 isAnalysisDemoFields37:= false
 var listDemoFieldsbody = []model.FieldModel{
 {
 FieldName: "msec",
 IsAnalysis: &isAnalysisDemoFields,
 },
 {
 FieldName: "access_log_topic_id",
 IsAnalysis: &isAnalysisDemoFields1,
 },
 {
 FieldName: "time_iso8601",
 IsAnalysis: &isAnalysisDemoFields2,
 },
 {
 FieldName: "log_ver",
 IsAnalysis: &isAnalysisDemoFields3,
 },
 {
 FieldName: "remote_addr",
 IsAnalysis: &isAnalysisDemoFields4,
 },
 {
 FieldName: "remote_port",
 IsAnalysis: &isAnalysisDemoFields5,
 },
 {
 FieldName: "status",
 IsAnalysis: &isAnalysisDemoFields6,
 },
 },
}
```

```
{
 fieldName: "request_method",
 isAnalysis: &isAnalysisDemoFields7,
},
{
 fieldName: "scheme",
 isAnalysis: &isAnalysisDemoFields8,
},
{
 fieldName: "host",
 isAnalysis: &isAnalysisDemoFields9,
},
{
 fieldName: "router_request_uri",
 isAnalysis: &isAnalysisDemoFields10,
},
{
 fieldName: "server_protocol",
 isAnalysis: &isAnalysisDemoFields11,
},
{
 fieldName: "request_length",
 isAnalysis: &isAnalysisDemoFields12,
},
{
 fieldName: "bytes_sent",
 isAnalysis: &isAnalysisDemoFields13,
},
{
 fieldName: "body_bytes_sent",
 isAnalysis: &isAnalysisDemoFields14,
},
{
 fieldName: "request_time",
 isAnalysis: &isAnalysisDemoFields15,
},
{
 fieldName: "upstream_status",
 isAnalysis: &isAnalysisDemoFields16,
},
{
 fieldName: "upstream_connect_time",
 isAnalysis: &isAnalysisDemoFields17,
},
{
 fieldName: "upstream_header_time",
 isAnalysis: &isAnalysisDemoFields18,
},
{
 fieldName: "upstream_response_time",
 isAnalysis: &isAnalysisDemoFields19,
},
{
 fieldName: "upstream_addr",
 isAnalysis: &isAnalysisDemoFields20,
},
{
 fieldName: "http_user_agent",
 isAnalysis: &isAnalysisDemoFields21,
},
{
 fieldName: "http_referer",
 isAnalysis: &isAnalysisDemoFields22,
},
{
 fieldName: "http_x_forwarded_for",
 isAnalysis: &isAnalysisDemoFields23,
},
{
```

```
 FieldName: "lb_name",
 IsAnalysis: &isAnalysisDemoFields24,
 },
 {
 FieldName: "listener_name",
 IsAnalysis: &isAnalysisDemoFields25,
 },
 {
 FieldName: "listener_id",
 IsAnalysis: &isAnalysisDemoFields26,
 },
 {
 FieldName: "pool_name",
 IsAnalysis: &isAnalysisDemoFields27,
 },
 {
 FieldName: "member_name",
 IsAnalysis: &isAnalysisDemoFields28,
 },
 {
 FieldName: "tenant_id",
 IsAnalysis: &isAnalysisDemoFields29,
 },
 {
 FieldName: "eip_address",
 IsAnalysis: &isAnalysisDemoFields30,
 },
 {
 FieldName: "eip_port",
 IsAnalysis: &isAnalysisDemoFields31,
 },
 {
 FieldName: "upstream_addr_priv",
 IsAnalysis: &isAnalysisDemoFields32,
 },
 {
 FieldName: "certificate_id",
 IsAnalysis: &isAnalysisDemoFields33,
 },
 {
 FieldName: "ssl_protocol",
 IsAnalysis: &isAnalysisDemoFields34,
 },
 {
 FieldName: "ssl_cipher",
 IsAnalysis: &isAnalysisDemoFields35,
 },
 {
 FieldName: "sni_domain_name",
 IsAnalysis: &isAnalysisDemoFields36,
 },
 {
 FieldName: "tcpinfo_rtt",
 IsAnalysis: &isAnalysisDemoFields37,
 },
}
quickAnalysisStructConfig:= false
request.Body = &model.StructConfig{
 QuickAnalysis: &quickAnalysisStructConfig,
 TagFields: &listTagFieldsbody,
 DemoFields: &listDemoFieldsbody,
 TemplateType: model.GetStructConfigTemplateTypeEnum().BUILT_IN,
 TemplateName: "ELB",
 TemplateId: "",
 LogStreamId: "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 LogGroupId: "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
}
response, err := client.UpdateStructConfig(request)
if err == nil {
```

```
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

- 修改VPC系统模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateStructConfigRequest{}
 isAnalysisTagFields := true
 var listTagFieldsbody = []model.FieldModel{
 {
 FieldName: "hostIP",
 IsAnalysis: &isAnalysisTagFields,
 },
 }
 var listDemoFieldsbody = []model.FieldModel{
 {
 FieldName: "version",
 },
 {
 FieldName: "project_id",
 },
 {
 FieldName: "interface_id",
 },
 {
 FieldName: "srcaddr",
 },
 {
 FieldName: "dstaddr",
 },
 {
 FieldName: "srcport",
 },
 {
 FieldName: "dstport",
 },
 },
}
```

```
{
 FileName: "protocol",
},
{
 FileName: "packets",
},
{
 FileName: "bytes",
},
{
 FileName: "start",
},
{
 FileName: "end",
},
{
 FileName: "action",
},
{
 FileName: "log_status",
},
}
quickAnalysisStructConfig:= false
request.Body = &model.StructConfig{
 QuickAnalysis: &quickAnalysisStructConfig,
 TagFields: &listTagFieldsbody,
 DemoFields: &listDemoFieldsbody,
 TemplateType: model.GetStructConfigTemplateTypeEnum().BUILT_IN,
 TemplateName: "VPC",
 TemplateId: "",
 LogStreamId: "b4d56d47-b4c4-453e-9047-xxxxxxxxxxxx",
 LogGroupId: "17f23e52-a23d-46e0-8bc5-xxxxxxxxxxxx",
}
response, err := client.UpdateStructConfig(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 201 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.8.3 删除结构化配置

### 功能介绍

该接口用于删除指定日志流下的结构化配置。

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v2/{project\_id}/lts/struct/template

表 6-385 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-386 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-387 请求 Body 参数

| 参数 | 是否必选 | 参数类型   | 描述                                                           |
|----|------|--------|--------------------------------------------------------------|
| id | 是    | String | 结构化规则ID<br>缺省值: None<br>最小长度: <b>106</b><br>最大长度: <b>106</b> |

## 响应参数

状态码: 200

表 6-388 响应 Body 参数

| 参数         | 参数类型   | 描述                                |
|------------|--------|-----------------------------------|
| error_code | String | SVCSTG.ALS.200200                 |
| error_msg  | String | delete struct config successfully |

状态码: 400

表 6-389 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值: <ul style="list-style-type: none"><li>• <b>SVCSTG.ALS.200201</b></li><li>• <b>SVCSTG.ALS.200203</b></li></ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值: <ul style="list-style-type: none"><li>• <b>structConfigId is error</b></li><li>• <b>delete redis failed.</b></li></ul> |

状态码: 401

表 6-390 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 状态码：403

表 6-391 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 状态码：500

表 6-392 响应 Body 参数

| 参数         | 参数类型   | 描述                                                             |
|------------|--------|----------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0202</b>                              |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Failed to query lts struct log</b> |

## 请求示例

删除当前ID的结构化配置

DELETE https://{endpoint}/v2/{project\_id}/lts/struct/template

```
/v2/{project_id}/lts/struct/template
{
 "id": "2a473356cca5487f8373be891bffc1cf_8a75b77d-7d72-4d7e-8c50-
a24562cf8b0b_fd5e1a7c-7412-475d-a013-8891d539574e"
}
```

## 响应示例

### 状态码：200

请求响应成功, 成功删除结构化配置。

```
{
 "id": "xxxxxx"
}
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求, 不要重试该请求。

```
{
 "errorCode": "LTS.0612",
 "errorMessage": "timee fieldType is error"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

### 状态码：500

InternalServerError。

表明服务端能被请求访问到, 但是服务内部出错。

```
{
 "error_code": "LTS.0010",
 "error_msg": "Internal Server Error"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

删除当前ID的结构化配置

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class DeleteStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 DeleteStructTemplateRequest request = new DeleteStructTemplateRequest();
 try {
 DeleteStructTemplateResponse response = client.deleteStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

### 删除当前ID的结构化配置

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")
```

```
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = DeleteStructTemplateRequest()
 response = client.delete_struct_template(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 删除当前ID的结构化配置

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.DeleteStructTemplateRequest{}
 response, err := client.DeleteStructTemplate(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                                  |
|-----|-----------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功删除结构化配置。                                                                                  |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。                                                      |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                     |
| 403 | Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                     |
| 503 | ServiceUnavailable。被请求的服务无效, 服务不可用。                                                                 |

## 错误码

请参见[错误码](#)。

## 6.8.4 查询结构化配置

### 功能介绍

该接口用于查询指定日志流下的结构化配置内容。

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/lts/struct/template

表 6-393 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

表 6-394 Query 参数

| 参数          | 是否必选 | 参数类型   | 描述                                                                                                                                   |
|-------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------|
| logGroupId  | 是    | String | 日志组ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：36<br>最大长度：36 |
| logStreamId | 是    | String | 日志流ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：36<br>最大长度：36 |

## 请求参数

表 6-395 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-396 响应 Body 参数

| 参数         | 参数类型                                                   | 描述      |
|------------|--------------------------------------------------------|---------|
| demoFields | Array of <a href="#">StructFieldInfoReturn</a> objects | 结构化字段   |
| tagFields  | Array of <a href="#">tag_fields_info</a> objects       | 关键词详细信息 |

| 参数           | 参数类型                                 | 描述      |
|--------------|--------------------------------------|---------|
| demoLog      | String                               | 示例日志    |
| demoLabel    | String                               | 参考日志属性  |
| id           | String                               | 结构化配置id |
| logGroupId   | String                               | 日志组ID   |
| rule         | ShowStructTemplateRule object        | 结构化方式   |
| cluster_info | ShowStructTemplateClusterInfo object | kafka信息 |
| logStreamId  | String                               | 日志流ID   |
| projectId    | String                               | 项目ID    |
| templateName | String                               | 模板名称    |
| regex        | String                               | 正则表达式   |

表 6-397 StructFieldInfoReturn

| 参数         | 参数类型    | 描述     |
|------------|---------|--------|
| fieldName  | String  | 字段名称   |
| type       | String  | 字段数据类型 |
| content    | String  | 字段内容   |
| isAnalysis | Boolean | 是否解析   |
| index      | Integer | 字段序号   |

表 6-398 tag\_fields\_info

| 参数         | 参数类型    | 描述   |
|------------|---------|------|
| fieldName  | String  | 字段名称 |
| type       | String  | 字段类型 |
| content    | String  | 内容   |
| isAnalysis | Boolean | 是否解析 |
| index      | Integer | 字段序号 |

表 6-399 ShowStructTemplateRule

| 参数    | 参数类型   | 描述    |
|-------|--------|-------|
| param | String | 结构化参数 |
| type  | String | 结构化类型 |

表 6-400 ShowStructTemplateclusterInfo

| 参数                      | 参数类型    | 描述               |
|-------------------------|---------|------------------|
| cluster_name            | String  | kafka集群名称        |
| kafka_bootstrap_servers | String  | kafka集群的服务器地址    |
| kafka_ssl_enable        | Boolean | kafka是否开启ssl加密认证 |

状态码：400

表 6-401 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                                                     |
|------------|--------|----------------------------------------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>SVCSTG.ALS.200201</li> <li>LTS.0208</li> </ul>                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Query Param is error</li> <li>The log stream does not existed</li> </ul> |

状态码：401

表 6-402 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                         |
|------------|--------|----------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul> |

| 参数        | 参数类型   | 描述                                                                                         |
|-----------|--------|--------------------------------------------------------------------------------------------|
| error_msg | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

**状态码：403**

表 6-403 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

**状态码：500**

表 6-404 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                      |
|------------|--------|---------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0202</li> </ul>                              |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Failed to query lts struct log</li> </ul> |

## 请求示例

查询当前租户下指定日志流下的结构化配置内容。请求参数为路径参数，不提供请求示例。

## 响应示例

**状态码：200**

请求响应成功, 成功获取结构化配置。

```
{
 "demoFields": [{
 "content": "100.19.10.178",
```

```
"fieldName": "authority",
"index": 0,
"isAnalysis": true,
"type": "string"
}, {
"content": "0",
"fieldName": "bytes_received",
"index": 0,
"isAnalysis": true,
"type": "string"
}, {
"content": "1127",
"fieldName": "bytes_sent",
"index": 0,
"isAnalysis": true,
"type": "string"
}
}]
}
```

**状态码：400**

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
"errorCode": "SVCSTG.ALS.200201",
"errorMessage": "Query Param is error."
}
```

**状态码：401**

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
"error_code": "LTS.0414",
"error_msg": "Invalid token"
}
```

**状态码：403**

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
"error_code": "LTS.0001",
"error_msg": "Invalid projectId"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
"error_code": "LTS.0102",
"error_msg": "Query empty."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ShowStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ShowStructTemplateRequest request = new ShowStructTemplateRequest();
 try {
 ShowStructTemplateResponse response = client.showStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")
 projectId = "{project_id}"
```

```
credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = ShowStructTemplateRequest()
 response = client.show_struct_template(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ShowStructTemplateRequest{}
 response, err := client.ShowStructTemplate(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                                  |
|-----|-----------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功获取结构化配置。                                                                                  |
| 400 | BadRequest。非法请求。 建议根据error_msg直接修改该请求, 不要重试该请求。                                                     |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                     |
| 403 | Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                     |
| 503 | ServiceUnavailable。被请求的服务无效, 服务不可用。                                                                 |

## 错误码

请参见[错误码](#)。

### 6.8.5 查询结构化模板简略列表

#### 功能介绍

该接口用于查询结构化模板简略列表。

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v3/{project\_id}/lts/struct/customtemplate/list

表 6-405 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-406 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-407 响应 Body 参数

| 参数      | 参数类型                                                      | 描述          |
|---------|-----------------------------------------------------------|-------------|
| results | Array of <a href="#">BriefStructTemplateModel</a> objects | 结构化模板缩略信息列表 |

表 6-408 BriefStructTemplateModel

| 参数            | 参数类型   | 描述                                         |
|---------------|--------|--------------------------------------------|
| create_time   | Long   | 模板创建/更新时间<br>最小值：0<br>最大值：2000000000000000 |
| id            | String | 模板id<br>最小长度：36<br>最大长度：36                 |
| template_name | String | 模板名称<br>最小长度：1<br>最大长度：64                  |

| 参数            | 参数类型   | 描述                                                                                                                                                          |
|---------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| template_type | String | 结构化类型, 当前支<br>regex,json,split,nginx<br>枚举值:<br><ul style="list-style-type: none"> <li>• regex</li> <li>• json</li> <li>• split</li> <li>• nginx</li> </ul> |
| project_id    | String | 项目ID, 获取方式请参见: 获取账号<br>ID、项目ID、日志组ID、日志流ID<br>最小长度: 32<br>最大长度: 32                                                                                          |

状态码: 500

表 6-409 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                               |
|------------|--------|--------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>最小长度: 8<br>最大长度: 8                                                                       |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值:<br><ul style="list-style-type: none"> <li>• Internal Server Error</li> </ul> |

## 请求示例

获取当前租户的结构化模板简略列表

```
GET https://{endpoint}/v3/{project_id}/lts/struct/customtemplate/list
/v3/{project_id}/lts/struct/customtemplate/list
```

## 响应示例

状态码: 200

请求响应成功。

```
{
 "results": [{
 "create_time": 1632897983441,
 "id": "47629e46-287d-478c-8888-xxxxxxxxxxxx",
 "template_name": "jsonTemplate",
 "template_type": "json",
 "project_id": "2a473356cca5487f8373be89xxxxxxxx"
 }
]
```

```
]]
}
```

#### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2017",
 "error_msg": "Find struct template failed."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.8.6 查询结构化模板

### 功能介绍

该接口用于查询结构化模板。

说明：

单个用户每秒最多可调用50次该接口。

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v3/{project\_id}/lts/struct/customtemplate

表 6-410 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-411 Query 参数

| 参数 | 是否必选 | 参数类型   | 描述                                                                    |
|----|------|--------|-----------------------------------------------------------------------|
| id | 否    | String | 待查询模板id,非必填, 不传时返回项目下所有自定义结构化模板<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |

## 请求参数

表 6-412 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                             |
|--------------|------|--------|------------------------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: <b>1000</b><br>最大长度: <b>2000</b> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: <b>30</b><br>最大长度: <b>30</b>                   |

## 响应参数

状态码: 200

表 6-413 响应 Body 参数

| 参数      | 参数类型                                                 | 描述            |
|---------|------------------------------------------------------|---------------|
| results | Array of <a href="#">StructTemplateModel</a> objects | 查询的自定义结构化模板数组 |

表 6-414 StructTemplateModel

| 参数         | 参数类型   | 描述                                         |
|------------|--------|--------------------------------------------|
| project_id | String | 项目id<br>最小长度: <b>32</b><br>最大长度: <b>32</b> |

| 参数            | 参数类型                                | 描述                                                                                                                                                  |
|---------------|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| template_name | String                              | 模板名称<br>最小长度: 1<br>最大长度: 64                                                                                                                         |
| template_type | String                              | 模板类型, regex,json,split,nginx<br>枚举值:<br><ul style="list-style-type: none"> <li>• regex</li> <li>• json</li> <li>• split</li> <li>• nginx</li> </ul> |
| demo_log      | String                              | 示例日志<br>最小长度: 1<br>最大长度: 5000                                                                                                                       |
| demo_fields   | Array of <b>DemoField</b> objects   | 示例字段数组                                                                                                                                              |
| tag_fields    | Array of <b>TagFieldNew</b> objects | Tag字段数组                                                                                                                                             |
| rule          | <b>TemplateRule</b> object          | 结构化规则对象                                                                                                                                             |
| demo_label    | String                              | 示例日志标签<br>最小长度: 0<br>最大长度: 5000                                                                                                                     |
| create_time   | Long                                | 创建时间<br>最小值: 0<br>最大值: 20000000000000                                                                                                               |
| id            | String                              | 模板id<br>最小长度: 36<br>最大长度: 36                                                                                                                        |

表 6-415 DemoField

| 参数         | 参数类型   | 描述                          |
|------------|--------|-----------------------------|
| field_name | String | 字段名称<br>最小长度: 1<br>最大长度: 64 |

| 参数                | 参数类型    | 描述                                                                                                                                |
|-------------------|---------|-----------------------------------------------------------------------------------------------------------------------------------|
| content           | String  | 字段示例内容<br>最小长度：1<br>最大长度：5000                                                                                                     |
| type              | String  | 字段数据类型。<br>可选范围：string、long、float<br>枚举值： <ul style="list-style-type: none"><li>• string</li><li>• long</li><li>• float</li></ul> |
| is_analysis       | Boolean | 是否开启快速分析                                                                                                                          |
| index             | Integer | 手动正则及分隔符方式中字段序号<br>最小值：0<br>最大值：200                                                                                               |
| relation          | String  | 描述多层次json中字段间的层级关系<br>最小长度：0<br>最大长度：1000                                                                                         |
| user_defined_name | String  | json及nginx方式中字段自定义别名<br>最小长度：1<br>最大长度：64                                                                                         |

表 6-416 TagFieldNew

| 参数         | 参数类型   | 描述                                                                                                                                |
|------------|--------|-----------------------------------------------------------------------------------------------------------------------------------|
| field_name | String | 字段名称<br>最小长度：1<br>最大长度：64                                                                                                         |
| content    | String | 字段示例内容<br>最小长度：0<br>最大长度：5000                                                                                                     |
| type       | String | 字段数据类型。<br>可选范围：string、long、float<br>枚举值： <ul style="list-style-type: none"><li>• string</li><li>• long</li><li>• float</li></ul> |

| 参数          | 参数类型    | 描述                                           |
|-------------|---------|----------------------------------------------|
| is_analysis | Boolean | 是否开启快速分析                                     |
| index       | Integer | 序号, 从0开始<br>最小值: <b>0</b><br>最大值: <b>200</b> |

表 6-417 TemplateRule

| 参数   | 参数类型   | 描述                                                                                                                                                                                            |
|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | String | 结构化类型, 只支持<br>custom_regex,json,split,nginx<br>枚举值: <ul style="list-style-type: none"><li>• <b>custom_regex</b></li><li>• <b>json</b></li><li>• <b>split</b></li><li>• <b>nginx</b></li></ul> |



| 参数 | 参数类型 | 描述        |
|----|------|-----------|
|    |      | 最大长度：5000 |

状态码：500

表 6-418 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                        |
|------------|--------|-------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Internal Server Error</li></ul> |

## 请求示例

查询当前结构化模板详情

```
GET https://{endpoint}/v3/{project_id}/lts/struct/customtemplate?id=bc8e3f2c-87fe-4acd-8439-69cdf29251c1/v3/{project_id}/lts/struct/customtemplate?id=bc8e3f2c-87fe-4acd-8439-69cdf29251c1
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "results": [{
 "create_time": 1641258099551,
 "demo_fields": [{
 "content": "2022-01-03/14:52:28",
 "field_name": "field1",
 "index": 0,
 "is_analysis": true,
 "type": "string"
 }, {
 "content": "this",
 "field_name": "field2",
 "index": 1,
 "is_analysis": true,
 "type": "string"
 }, {
 "content": "log",
 "field_name": "field3",
 "index": 2,
 "is_analysis": false,
 "type": "string"
 }, {
 "content": "is",
 "field_name": "field4",
```

```
"index" : 3,
"is_analysis" : false,
"type" : "string"
}, {
 "content" : "Error",
 "field_name" : "field5",
 "index" : 4,
 "is_analysis" : false,
 "type" : "string"
}, {
 "content" : "NO",
 "field_name" : "field6",
 "index" : 5,
 "is_analysis" : false,
 "type" : "string"
}, {
 "content" : "13测试",
 "field_name" : "field7",
 "index" : 6,
 "is_analysis" : false,
 "type" : "string"
}, {
 "content" : "286",
 "field_name" : "field8",
 "index" : 7,
 "is_analysis" : false,
 "type" : "long"
}
],
"demo_log" : "2022-01-03/14:52:28 this log is Error NO 13测试 286",
"id" : "43a8cc7b-b632-4c36-a65d-8150e98219f1",
"project_id" : "2a473356cca5487f8373be89xxxxxxx",
"rule" : {
 "param" : "{ \"keyObject\" :
{ \\\"0\\\": \\\"field1\\\", \\\"1\\\": \\\"field2\\\", \\\"2\\\": \\\"field3\\\", \\\"3\\\": \\\"field4\\\", \\\"4\\\": \\\"field5\\\", \\\"5\\\": \\\"field6\\\", \\\"6\\\": \\\"field7\\\", \\\"7\\\": \\\"field8\\\"}, \\\"tokenizer\\\": \\\" \\\" }\",
 "type" : "split"
},
"demo_label" : "{}",
"tag_fields" : [{
 "content" : "172.16.10.69",
 "field_name" : "hostIP",
 "index" : 0,
 "is_analysis" : true,
 "type" : "string"
}],
"template_name" : "testSplit13",
"template_type" : "split"
}]
}
```

### 状态码: 500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2017",
 "error_msg" : "Find struct template failed."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.8.7 创建结构化配置（不推荐）

### 功能介绍

该接口用于创建指定日志流下的结构化配置。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/lts/struct/template

表 6-419 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                                            |
|------------|------|--------|---------------------------------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a> |

### 请求参数

表 6-420 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                  |
|--------------|------|--------|-----------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=utf8。                |
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a> |

表 6-421 请求 Body 参数

| 参数           | 是否必选 | 参数类型   | 描述                                          |
|--------------|------|--------|---------------------------------------------|
| log_group_id | 是    | String | 日志组ID<br>最小长度： <b>36</b><br>最大长度： <b>36</b> |

| 参数            | 是否必选 | 参数类型                               | 描述                                                                         |
|---------------|------|------------------------------------|----------------------------------------------------------------------------|
| log_stream_id | 是    | String                             | 日志流ID<br>最小长度：36<br>最大长度：36                                                |
| project_id    | 是    | String                             | 项目ID<br>最小长度：32<br>最大长度：32                                                 |
| content       | 是    | String                             | 填写一条示例日志。                                                                  |
| demo_fields   | 是    | Array of <b>DemoFields</b> objects | 示例字段列表。<br><b>说明</b><br>demo_fields中的字段顺序，须和content中的字段顺序保持一致。<br>最大个数：200 |
| parse_type    | 是    | String                             | 结构化类型参数（json、split、nginx、built_in、custom_regex）。                           |
| regex_rules   | 否    | String                             | 正则表达式（使用正则结构化规则时候需要）。<br>最大长度：5000                                         |
| layers        | 否    | Integer                            | 最大解析层数（使用JSON方式结构化需要）。<br>最大值：3                                            |
| tokenizer     | 否    | String                             | 分隔符（使用分隔符结构化时需要）。                                                          |
| log_format    | 否    | String                             | Nginx配置规则（使用分隔符结构化时需要）。                                                    |
| tag_fields    | 是    | Array of <b>TagField</b> objects   | tag字段列表（使用tag字段解析时需要，其中系统模板不支持使用tag字段）。<br>最大个数：200                        |
| rule          | 否    | <b>StructRule</b> object           | 结构化方式                                                                      |

表 6-422 DemoFields

| 参数        | 是否必选 | 参数类型   | 描述                        |
|-----------|------|--------|---------------------------|
| fieldName | 否    | String | 字段名称<br>最小长度：1<br>最大长度：50 |

| 参数              | 是否必选 | 参数类型    | 描述                                                             |
|-----------------|------|---------|----------------------------------------------------------------|
| userDefinedName | 否    | String  | 自定义别名(json, Nginx方式中按需添加)<br>最小长度: <b>1</b><br>最大长度: <b>64</b> |
| type            | 是    | String  | 字段数据类型。<br>可选范围: string、long、float                             |
| isAnalysis      | 否    | Boolean | 预留字段。                                                          |
| content         | 否    | String  | 字段内容。<br><b>说明</b><br>在系统模板中, 此参数为demoLog。                     |
| index           | 否    | Integer | 序号                                                             |

表 6-423 TagField

| 参数         | 是否必选 | 参数类型    | 描述                                         |
|------------|------|---------|--------------------------------------------|
| fieldName  | 是    | String  | 字段名称<br>最小长度: <b>1</b><br>最大长度: <b>50</b>  |
| content    | 否    | String  | 示例字段。<br>最小长度: <b>1</b><br>最大长度: <b>64</b> |
| type       | 是    | String  | 字段数据类型。可选范围:<br>string、long、float          |
| isAnalysis | 否    | Boolean | 预留字段。                                      |

表 6-424 StructRule

| 参数    | 是否必选 | 参数类型   | 描述       |
|-------|------|--------|----------|
| type  | 是    | String | 内置系统模板类型 |
| param | 是    | String | 系统模板名称   |

## 响应参数

状态码: 200

表 6-425 响应 Body 参数

| 参数 | 参数类型   | 描述 |
|----|--------|----|
| -  | String |    |

状态码：400

表 6-426 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                         |
|------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• SVCSTG.ALS.200201</li> <li>• SVCSTG.ALS.200340</li> <li>• SVCSTG.ALS.200332</li> <li>• SVCSTG.ALS.200209</li> <li>• SVCSTG.ALS.200.339</li> </ul>                                                                                                                                                    |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• parseType is invalid/ smart extra failed./ The number of demo_fields and rules is not equal</li> <li>• {字段名/别名} is repeated with internal fields</li> <li>• userDefinedName repeat./ field name repeat</li> <li>• check logContent type error</li> <li>• nginx log_format is error</li> </ul> |

状态码：401

表 6-427 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                        |
|------------|--------|---------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• LTS.0403</li> </ul> |

| 参数        | 参数类型   | 描述                                                |
|-----------|--------|---------------------------------------------------|
| error_msg | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

**状态码：403**

表 6-428 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

**状态码：500**

表 6-429 响应 Body 参数

| 参数         | 参数类型   | 描述                                                             |
|------------|--------|----------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0500</b>                              |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Failed to query lts struct log</b> |

## 请求示例

- VPC:

POST https://{endpoint}/v2/{project\_id}/lts/struct/template

```
{
 "content": "1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd
192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK",
 "demo_fields": [{
 "fieldName": "version",
 "content": 1,
 "type": "long",
 "isAnalysis": true
 }
]
```

```
}, {
 "fieldName": "project_id",
 "content": "5f67944957444bd6bb4fe3b367de8f3d",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "interface_id",
 "content": "1d515d18-1b36-47dc-a983-bd6512aed4bd",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "srcaddr",
 "content": "192.168.0.154",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "dstaddr",
 "content": "192.168.3.25",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "srcport",
 "content": "38929",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "dstport",
 "content": "53",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "protocol",
 "content": "17",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "packets",
 "content": "1",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "bytes",
 "content": "96",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "start",
 "content": "1548752136",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "end",
 "content": "1548752736",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "action",
 "content": "ACCEPT",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "log_status",
 "content": "OK",
 "type": "string",
 "isAnalysis": true
}],
"rule": {
 "type": "built_in",
 "param": "VPC"
}
```

```
},
"tag_fields": [{
 "fieldName": "hostIP",
 "content": "192.168.2.134",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "hostName",
 "content": "ecs-icest",
 "type": "string",
 "isAnalysis": true
}],
"log_group_id": "925a750-e0f3-4fe9-a046-a04af676xxxx",
"log_stream_id": "7e432db8-9dad-4723-a4b1-fdabf712xxxx",
"parse_type": " ",
"project_id": "2a473356cca5487f8373be891bffxxxx"
}
```

- ELB:

POST https://{endpoint}/v2/{project\_id}/lts/struct/template

```
{
 "content": "1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac 2020-07-14T19:57:36+08:00
elb_01 192.0.0.0:88888 200 \\\"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\\\" 233 293 138 0.001 \\\"200\\\" \\\"0.000\\\" \\\"0.001\\\" \\\"0.001\\\"
\\\"100.0.0.0:9999\\\" \\\"lua-resty-http/0.14 (Lua) ngx_lua/10000\\\" \\\"-\\\" \\\"-\\\" loadbalancer_edsaee-4c9c-
b467-5b8126b2f7f7dsa listener_6077809b-913f-466d-a96c-376f08882d5d
08cc2b3f68aa4ddddd1e6a90ddddd1688348a4480 pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \\\"-\\\"
2fb78dsadadq1213das1121dab146ad3cb0 -:80 \\\"101.0.0.0:10000\\\" - - - - 9739\",
 "demo_fields": [{
 "fieldName": "msec",
 "content": "1594727856.337",
 "type": "float",
 "isAnalysis": true
 }, {
 "fieldName": "access_log_topic_id",
 "content": "e7c37d97-e922-457c-bbf3-dsadeqac",
 "type": "string",
 "isAnalysis": true
 }, {
 "fieldName": "time_iso8601",
 "content": "2020-07-14T19:57:36+08:00",
 "type": "string",
 "isAnalysis": true
 }, {
 "fieldName": "log_ver",
 "content": "elb_01",
 "type": "string",
 "isAnalysis": true
 }, {
 "fieldName": "remote_addr",
 "content": "192.0.0.0",
 "type": "string",
 "isAnalysis": true
 }, {
 "fieldName": "remote_port",
 "content": "88888",
 "type": "long",
 "isAnalysis": true
 }, {
 "fieldName": "status",
 "content": "200",
 "type": "long",
 "isAnalysis": true
 }, {
 "fieldName": "request_method",
 "content": "GET",
 "type": "string",
 "isAnalysis": true
 }
]
```

```
}, {
 "fieldName": "scheme",
 "content": "http",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "host",
 "content": "prod.sss.ads.sg2.aaa",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "router_request_uri",
 "content": "loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "server_protocol",
 "content": "HTTP/1.1",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "request_length",
 "content": "233",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "bytes_sent",
 "content": "293",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "body_bytes_sent",
 "content": "138",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "request_time",
 "content": "0.001",
 "type": "float",
 "isAnalysis": true
}, {
 "fieldName": "upstream_status",
 "content": "200",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "upstream_connect_time",
 "content": "0.000",
 "type": "float",
 "isAnalysis": true
}, {
 "fieldName": "upstream_header_time",
 "content": "0.001",
 "type": "float",
 "isAnalysis": true
}, {
 "fieldName": "upstream_response_time",
 "content": "0.001",
 "type": "float",
 "isAnalysis": true
}, {
 "fieldName": "upstream_addr",
 "content": "100.0.0.0:9999",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "http_user_agent",
```

```
"content": "lua-resty-http/0.14 (Lua) ngx_lua/10000",
"type": "string",
"isAnalysis": true
}, {
 "fieldName": "http_referer",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "http_x_forwarded_for",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "lb_name",
 "content": "loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "listener_name",
 "content": "listener_6077809b-913f-466d-a96c-376f08882d5d",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "listener_id",
 "content": "08cc2b3f68aa4ddd1e6a90ddd1688348a4480",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "pool_name",
 "content": "pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "member_name",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "tenant_id",
 "content": "2fb78dsadadq1213das1121dab146ad3cb0",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "eip_address",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "eip_port",
 "content": "80",
 "type": "long",
 "isAnalysis": true
}, {
 "fieldName": "upstream_addr_priv",
 "content": "101.0.0.0:10000",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "certificate_id",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "ssl_protocol",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {

```

```
{
 "fieldName": "ssl_cipher",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "sni_domain_name",
 "content": "-",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "tcpinfo_rtt",
 "content": "9739",
 "type": "long",
 "isAnalysis": true
}],
"rule": {
 "type": "built_in",
 "param": "ELB"
},
"tag_fields": [{
 "fieldName": "hostIP",
 "content": "192.168.2.134",
 "type": "string",
 "isAnalysis": true
}, {
 "fieldName": "hostName",
 "content": "ecs-icest",
 "type": "string",
 "isAnalysis": true
}],
"log_group_id": "925a750-e0f3-4fe9-a046-a04af676xxxx",
"log_stream_id": "7e432db8-9dad-4723-a4b1-fdabf712xxxx",
"parse_type": "",
"project_id": "2a473356cca5487f8373be891bffxxxx"
}
```

## 响应示例

### 状态码：200

请求响应成功, 成功创建结构化配置。

```
{
 "2a473356cca5487f8373be891bff1cf_8a75b77d-7d72-4d7e-8c50-a24562cf8b0b_fd5e1a7c-7412-475d-a013-8891d539574e"
}
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode": "LTS.0612",
 "errorMessage": "timee fieldType is error"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

### 状态码：403

Forbidden。

请求被拒绝访问。

返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code" : "LTS.0001",
 "error_msg" : "Invalid projectId"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- VPC:

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateStructTemplateRequest request = new CreateStructTemplateRequest();
 LtsStructTemplateInfo body = new LtsStructTemplateInfo();
 Rule rulebody = new Rule();
 rulebody.withType("built_in")
 .withParam("VPC");
 List<TagField> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new TagField()
 .withFieldName("hostIP")
 .withType("string")
 .withContent("192.168.2.134")
 .withIsAnalysis(true)
);
 }
}
```

```
);
listbodyTagFields.add(
 new TagField()
 .withFieldName("hostName")
 .withType("string")
 .withContent("ecs-icetest")
 .withIsAnalysis(true)
);
List<StructFieldInfo> listbodyDemoFields = new ArrayList<>();
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1")
 .withFieldName("version")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("5f67944957444bd6bb4fe3b367de8f3d")
 .withFieldName("project_id")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1d515d18-1b36-47dc-a983-bd6512aed4bd")
 .withFieldName("interface_id")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("192.168.0.154")
 .withFieldName("srcaddr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("192.168.3.25")
 .withFieldName("dstaddr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("38929")
 .withFieldName("srcport")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("53")
 .withFieldName("dstport")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("17")
 .withFieldName("protocol")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1")
);
```

```
 .withFieldName("packets")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("96")
 .withFieldName("bytes")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1548752136")
 .withFieldName("start")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1548752736")
 .withFieldName("end")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("ACCEPT")
 .withFieldName("action")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("OK")
 .withFieldName("log_status")
 .withType("string")
);
 body.withRule(rulebody);
 body.withProjectId("2a473356cca5487f8373be891bffxxxx");
 body.withLogStreamId("7e432db8-9dad-4723-a4b1-fdabf712xxxx");
 body.withParseType(LtsStructTemplateInfo.ParseTypeEnum.fromValue(""));
 body.withLogGroupId("925a750-e0f3-4fe9-a046-a04af676xxxx");
 body.withContent("1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-
bd6512aed4bd 192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK");
 body.withTagFields(listbodyTagFields);
 body.withDemoFields(listbodyDemoFields);
 request.withBody(body);
 try {
 CreateStructTemplateResponse response = client.createStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

- **ELB:**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
```

```
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateStructTemplateRequest request = new CreateStructTemplateRequest();
 LtsStructTemplateInfo body = new LtsStructTemplateInfo();
 Rule rulebody = new Rule();
 rulebody.withType("built_in")
 .withParam("ELB");
 List<TagField> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new TagField()
 .withFieldName("hostIP")
 .withType("string")
 .withContent("192.168.2.134")
 .withIsAnalysis(true)
);
 listbodyTagFields.add(
 new TagField()
 .withFieldName("hostName")
 .withType("string")
 .withContent("ecs-ictest")
 .withIsAnalysis(true)
);
 List<StructFieldInfo> listbodyDemoFields = new ArrayList<>();
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("1594727856.337")
 .withFieldName("msec")
 .withType("float")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("e7c37d97-e922-457c-bbf3-dsadeqac")
 .withFieldName("access_log_topic_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()

```

```
.withIsAnalysis(true)
.withContent("2020-07-14T19:57:36+08:00")
.withFieldName("time_iso8601")
.withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("elb_01")
 .withFieldName("log_ver")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("192.0.0.0")
 .withFieldName("remote_addr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("88888")
 .withFieldName("remote_port")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("200")
 .withFieldName("status")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("GET")
 .withFieldName("request_method")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("http")
 .withFieldName("scheme")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("prod.sss.ads.sg2.aaa")
 .withFieldName("host")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("/location?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsad&beyla_id=w
qeb123ndadsa233ddada")
 .withFieldName("router_request_uri")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("HTTP/1.1")
 .withFieldName("server_protocol")
 .withType("string")
);
```

```
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("233")
 .withFieldName("request_length")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("293")
 .withFieldName("bytes_sent")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("138")
 .withFieldName("body_bytes_sent")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("0.001")
 .withFieldName("request_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("200")
 .withFieldName("upstream_status")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("0.000")
 .withFieldName("upstream_connect_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("0.001")
 .withFieldName("upstream_header_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("0.001")
 .withFieldName("upstream_response_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("100.0.0.0:9999")
 .withFieldName("upstream_addr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("lua-resty-http/0.14 (Lua ngx_lua/10000)")
 .withFieldName("http_user_agent")
);
```

```
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("http_referer")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("http_x_forwarded_for")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa")
 .withFieldName("lb_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("listener_6077809b-913f-466d-a96c-376f08882d5d")
 .withFieldName("listener_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("08cc2b3f68aa4dddd1e6a90ddd1688348a4480")
 .withFieldName("listener_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b")
 .withFieldName("pool_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("member_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("2fb78dsadadq1213das1121dab146ad3cb0")
 .withFieldName("tenant_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("eip_address")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("80")
);
```

```
 .withFieldName("eip_port")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("101.0.0.0:10000")
 .withFieldName("upstream_addr_priv")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("certificate_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("ssl_protocol")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("ssl_cipher")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("-")
 .withFieldName("sni_domain_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withIsAnalysis(true)
 .withContent("9739")
 .withFieldName("tcpinfo_rtt")
 .withType("long")
);
 body.withRule(rulebody);
 body.withProjectId("2a473356cca5487f8373be891bffxxx");
 body.withLogStreamId("7e432db8-9dad-4723-a4b1-fdabf712xxx");
 body.withParseType(LtsStructTemplateInfo.ParseTypeEnum.fromValue(""));
 body.withLogGroupId("925a750-e0f3-4fe9-a046-a04af676xxx");
 body.withContent("1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac
2020-07-14T19:57:36+08:00 elb_01 192.0.0.0:88888 200 \"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\" 233 293 138 0.001 \"200\" \"0.000\" \"0.001\" \"0.001\" \"100.0.0.0:9999\"
\"lua-resty-http/0.14 (Lua) ngx_lua/10000\" \"-\" \"-\" loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa
listener_6077809b-913f-466d-a96c-376f08882d5d 08cc2b3f68aa4ddd1e6a90ddd1688348a4480
pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \"-\" 2fb78dsadadq1213das1121dab146ad3cb0 -:80
\"101.0.0.0:10000\" - - - - 9739");
 body.withTagFields(listbodyTagFields);
 body.withDemoFields(listbodyDemoFields);
 request.withBody(body);
 try {
 CreateStructTemplateResponse response = client.createStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

- VPC:

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudskcls.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudskcls.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateStructTemplateRequest()
 rulebody = Rule(
 type="built_in",
 param="VPC"
)
 listTagFieldsbody = [
 TagField(
 field_name="hostIP",
 type="string",
 content="192.168.2.134",
 is_analysis=True
),
 TagField(
 field_name="hostName",
 type="string",
 content="ecs-ictest",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 StructFieldInfo(
 is_analysis=True,
 content="1",
 field_name="version",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="5f67944957444bd6bb4fe3b367de8f3d",
 field_name="project_id",
```

```
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="1d515d18-1b36-47dc-a983-bd6512aed4bd",
 field_name="interface_id",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="192.168.0.154",
 field_name="srcaddr",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="192.168.3.25",
 field_name="dstaddr",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="38929",
 field_name="srcport",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="53",
 field_name="dstport",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="17",
 field_name="protocol",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="1",
 field_name="packets",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="96",
 field_name="bytes",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="1548752136",
 field_name="start",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="1548752736",
 field_name="end",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="ACCEPT",
 field_name="action",
 type="string"
),
 StructFieldInfo(
```

```
 is_analysis=True,
 content="OK",
 field_name="log_status",
 type="string"
)
]
request.body = LtsStructTemplateInfo(
 rule=rulebody,
 project_id="2a473356cca5487f8373be891bffxxx",
 log_stream_id="7e432db8-9dad-4723-a4b1-fdabf712xxxx",
 parse_type=" ",
 log_group_id="925a750-e0f3-4fe9-a046-a04af676xxxx",
 content="1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd
192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK",
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody
)
response = client.create_struct_template(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

- **ELB:**

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateStructTemplateRequest()
 rulebody = Rule(
 type="built_in",
 param="ELB"
)
 listTagFieldsbody = [
 TagField(
 field_name="hostIP",
 type="string",
 content="192.168.2.134",
 is_analysis=True
),
 TagField(
 field_name="hostName",
 type="string",
 content="ecs-ictest",
 is_analysis=True
)
]
```

```
)
]
listDemoFieldsbody = [
 StructFieldInfo(
 is_analysis=True,
 content="1594727856.337",
 field_name="msec",
 type="float"
),
 StructFieldInfo(
 is_analysis=True,
 content="e7c37d97-e922-457c-bbf3-dsadeqac",
 field_name="access_log_topic_id",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="2020-07-14T19:57:36+08:00",
 field_name="time_iso8601",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="elb_01",
 field_name="log_ver",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="192.0.0.0",
 field_name="remote_addr",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="88888",
 field_name="remote_port",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="200",
 field_name="status",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="GET",
 field_name="request_method",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="http",
 field_name="scheme",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="prod.sss.ads.sg2.aaa",
 field_name="host",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada",
 field_name="router_request_uri",
```

```
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="HTTP/1.1",
 field_name="server_protocol",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="233",
 field_name="request_length",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="293",
 field_name="bytes_sent",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="138",
 field_name="body_bytes_sent",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="0.001",
 field_name="request_time",
 type="float"
),
 StructFieldInfo(
 is_analysis=True,
 content="200",
 field_name="upstream_status",
 type="long"
),
 StructFieldInfo(
 is_analysis=True,
 content="0.000",
 field_name="upstream_connect_time",
 type="float"
),
 StructFieldInfo(
 is_analysis=True,
 content="0.001",
 field_name="upstream_header_time",
 type="float"
),
 StructFieldInfo(
 is_analysis=True,
 content="0.001",
 field_name="upstream_response_time",
 type="float"
),
 StructFieldInfo(
 is_analysis=True,
 content="100.0.0.0:9999",
 field_name="upstream_addr",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="lua-resty-http/0.14 (Lua) ngx_lua/10000",
 field_name="http_user_agent",
 type="string"
),
 StructFieldInfo(
```

```
is_analysis=True,
content="-",
field_name="http_referer",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="-",
field_name="http_x_forwarded_for",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa",
field_name="lb_name",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="listener_6077809b-913f-466d-a96c-376f08882d5d",
field_name="listener_name",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="08cc2b3f68aa4dddd1e6a90ddd1688348a4480",
field_name="listener_id",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b",
field_name="pool_name",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="-",
field_name="member_name",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="2fb78dsadq1213das1121dab146ad3cb0",
field_name="tenant_id",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="-",
field_name="eip_address",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="80",
field_name="eip_port",
type="long"
),
StructFieldInfo(
is_analysis=True,
content="101.0.0.0:10000",
field_name="upstream_addr_priv",
type="string"
),
StructFieldInfo(
is_analysis=True,
content="-",
field_name="certificate_id",
```

```
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="-",
 field_name="ssl_protocol",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="-",
 field_name="ssl_cipher",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="-",
 field_name="sni_domain_name",
 type="string"
),
 StructFieldInfo(
 is_analysis=True,
 content="9739",
 field_name="tcpinfo_rtt",
 type="long"
)
]
request.body = LtsStructTemplateInfo(
 rule=rulebody,
 project_id="2a473356cca5487f8373be891bffxxxx",
 log_stream_id="7e432db8-9dad-4723-a4b1-fdabf712xxxx",
 parse_type=" ",
 log_group_id="925a750-e0f3-4fe9-a046-a04af676xxxx",
 content="1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac 2020-07-14T19:57:36+08:00
elb_01 192.0.0.0:88888 200 \"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\" 233 293 138 0.001 \"200\" \"0.000\" \"0.001\" \"0.001\" \"100.0.0.0:9999\"
\"lua-resty-http/0.14 (Lua) ngx_lua/10000\" \"-\" \"-\" loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa
listener_6077809b-913f-466d-a96c-376f08882d5d 08cc2b3f68aa4ddddd1e6a90ddddd1688348a4480
pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \"-\" 2fb78dsadadq1213das1121dab146ad3cb0 -:80
\"101.0.0.0:10000\" - - - - 9739\",
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody
)
response = client.create_struct_template(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- VPC:
- ELB:

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                                          |
|-----|-------------------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功创建结构化配置。                                                                                          |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。                                                              |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                             |
| 403 | Forbidden。<br>请求被拒绝访问。<br>返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                             |
| 503 | ServiceUnavailable。<br>被请求的服务无效, 服务不可用。                                                                     |

## 错误码

请参见[错误码](#)。

### 6.8.8 修改结构化配置（不推荐）

#### 功能介绍

该接口用于修改指定日志流下的结构化配置, 由于参数较多且构建复杂, 建议您使用[8.9.2-修改结构化配置（推荐）](#)。

#### 调用方法

请参见[如何调用API](#)。

#### URI

PUT /v2/{project\_id}/lts/struct/template

表 6-430 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                         |
|------------|------|--------|------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a> |

## 请求参数

表 6-431 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                   |
|--------------|------|--------|------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。               |

表 6-432 请求 Body 参数

| 参数            | 是否必选 | 参数类型                               | 描述                                                                           |
|---------------|------|------------------------------------|------------------------------------------------------------------------------|
| log_group_id  | 是    | String                             | 日志组ID<br>最小长度: <b>36</b><br>最大长度: <b>36</b>                                  |
| log_stream_id | 是    | String                             | 日志流ID<br>最小长度: <b>36</b><br>最大长度: <b>36</b>                                  |
| project_id    | 是    | String                             | 项目ID<br>最小长度: <b>32</b><br>最大长度: <b>32</b>                                   |
| content       | 是    | String                             | 填写一条示例日志。                                                                    |
| demo_fields   | 是    | Array of <b>DemoFields</b> objects | 示例字段列表。<br><b>说明</b><br>demo_fields中的字段顺序, 须和content中的字段顺序保持一致。<br>最大个数: 200 |
| parse_type    | 是    | String                             | 结构化类型参数 ( json、split、nginx、built_in、custom_regex )。                          |
| regex_rules   | 否    | String                             | 正则表达式 ( 使用正则结构化规则时候需要 )。<br>最大长度: <b>5000</b>                                |
| layers        | 否    | Integer                            | 最大解析层数 ( 使用JSON方式结构化需要 )。<br>最大值: <b>3</b>                                   |
| tokenizer     | 否    | String                             | 分隔符 ( 使用分隔符结构化时候需要 )。                                                        |

| 参数         | 是否必选 | 参数类型                             | 描述                                                  |
|------------|------|----------------------------------|-----------------------------------------------------|
| log_format | 否    | String                           | Nginx配置规则（使用分隔符结构化时需要）。                             |
| tag_fields | 是    | Array of <b>TagField</b> objects | tag字段列表（使用tag字段解析时需要，其中系统模板不支持使用tag字段）。<br>最大个数：200 |
| rule       | 否    | <b>StructRule</b> object         | 结构化方式                                               |

表 6-433 DemoFields

| 参数              | 是否必选 | 参数类型    | 描述                                             |
|-----------------|------|---------|------------------------------------------------|
| fieldName       | 否    | String  | 字段名称<br>最小长度：1<br>最大长度：50                      |
| userDefinedName | 否    | String  | 自定义别名(json, Nginx方式中按需添加)<br>最小长度：1<br>最大长度：64 |
| type            | 是    | String  | 字段数据类型。<br>可选范围：string、long、float              |
| isAnalysis      | 否    | Boolean | 预留字段。                                          |
| content         | 否    | String  | 字段内容。<br><b>说明</b><br>在系统模板中，此参数为demoLog。      |
| index           | 否    | Integer | 序号                                             |

表 6-434 TagField

| 参数        | 是否必选 | 参数类型   | 描述                         |
|-----------|------|--------|----------------------------|
| fieldName | 是    | String | 字段名称<br>最小长度：1<br>最大长度：50  |
| content   | 否    | String | 示例字段。<br>最小长度：1<br>最大长度：64 |

| 参数         | 是否必选 | 参数类型    | 描述                            |
|------------|------|---------|-------------------------------|
| type       | 是    | String  | 字段数据类型。可选范围：string、long、float |
| isAnalysis | 否    | Boolean | 预留字段。                         |

表 6-435 StructRule

| 参数    | 是否必选 | 参数类型   | 描述       |
|-------|------|--------|----------|
| type  | 是    | String | 内置系统模板类型 |
| param | 是    | String | 系统模板名称   |

## 响应参数

状态码：200

表 6-436 响应 Body 参数

| 参数 | 参数类型   | 描述 |
|----|--------|----|
| -  | String |    |

状态码：400

表 6-437 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                                                                                                                                      |
|------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• SVCSTG.ALS.200201</li> <li>• SVCSTG.ALS.200340</li> <li>• SVCSTG.ALS.200332</li> <li>• SVCSTG.ALS.200209</li> <li>• SVCSTG.ALS.200.339</li> </ul> |

| 参数        | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>parseType is invalid/ smart extra failed./ The number of demo_fields and rules is not equal</b></li> <li>• <b>{字段名/别名} is repeated with internal fields</b></li> <li>• <b>userDefinedName repeat./ field name repeat</b></li> <li>• <b>check logContent type error</b></li> <li>• <b>nginx log_format is error</b></li> </ul> |

状态码：401

表 6-438 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                               |
|------------|--------|--------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>LTS.0403</b></li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>Invalid projectId</b></li> </ul> |

状态码：403

表 6-439 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                               |
|------------|--------|--------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>LTS.0403</b></li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• <b>Invalid projectId</b></li> </ul> |

**状态码：500**

**表 6-440 响应 Body 参数**

| 参数         | 参数类型   | 描述                                                             |
|------------|--------|----------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0202</b>                              |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Failed to query lts struct log</b> |

**状态码：503**

**表 6-441 响应 Body 参数**

| 参数 | 参数类型   | 描述 |
|----|--------|----|
| -  | String |    |

**请求示例**

- 修改VPC方式的结构化配置。

PUT https://{endpoint}/v2/{project\_id}/lts/struct/template

```
{
 "content": "1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd
192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK",
 "demo_fields": [{
 "fieldName": "version",
 "content": 1,
 "type": "long"
 }, {
 "fieldName": "project_id",
 "content": "5f67944957444bd6bb4fe3b367de8f3d",
 "type": "string"
 }, {
 "fieldName": "interface_id",
 "content": "1d515d18-1b36-47dc-a983-bd6512aed4bd",
 "type": "string"
 }, {
 "fieldName": "srcaddr",
 "content": "192.168.0.154",
 "type": "string"
 }, {
 "fieldName": "dstaddr",
 "content": "192.168.3.25",
 "type": "string"
 }, {
 "fieldName": "srcport",
 "content": "38929",
 "type": "long"
 }, {
 "fieldName": "dstport",
```

```
"content": "53",
"type": "long"
}, {
 "fieldName": "protocol",
 "content": "17",
 "type": "long"
}, {
 "fieldName": "packets",
 "content": "1",
 "type": "long"
}, {
 "fieldName": "bytes",
 "content": "96",
 "type": "long"
}, {
 "fieldName": "start",
 "content": "1548752136",
 "type": "long"
}, {
 "fieldName": "end",
 "content": "1548752736",
 "type": "long"
}, {
 "fieldName": "action",
 "content": "ACCEPT",
 "type": "string"
}, {
 "fieldName": "log_status",
 "content": "OK",
 "type": "string"
}],
"rule": {
 "type": "built_in",
 "param": "VPC"
},
"log_group_id": "",
"log_stream_id": "",
"project_id": "",
"parse_type": "",
"tag_fields": [{
 "fieldName": "hostIP",
 "content": "192.168.2.134",
 "type": "string",
 "isAnalysis": true
}]
}]
}
```

- 修改ELB方式的结构化配置。

PUT [https://{endpoint}/v2/{project\\_id}/lts/struct/template](https://{endpoint}/v2/{project_id}/lts/struct/template)

```
{
 "content": "1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac 2020-07-14T19:57:36+08:00
elb_01 192.0.0.0:88888 200 \\\"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadasadsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\\\" 233 293 138 0.001 \\\"200\\\" \\\"0.000\\\" \\\"0.001\\\" \\\"0.001\\\"
\\\"100.0.0.0:9999\\\" \\\"lua-resty-http/0.14 (Lua) ngx_lua/10000\\\" \\\"-\\\" \\\"-\\\" loadbalancer_edsaee-4c9c-
b467-5b8126b2f7f7dsa listener_6077809b-913f-466d-a96c-376f08882d5d
08cc2b3f68aa4ddd1e6a90ddd1688348a4480 pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \\\"-\\\"
2fb78dsadadq1213das1121dab146ad3cb0 -80 \\\"101.0.0.0:10000\\\" - - - - 9739\",
 "demo_fields": [{
 "fieldName": "msec",
 "content": "1594727856.337",
 "type": "float"
 }, {
 "fieldName": "access_log_topic_id",
 "content": "e7c37d97-e922-457c-bbf3-dsadeqac",
 "type": "string"
 }, {
 "fieldName": "time_iso8601",
 "content": "2020-07-14T19:57:36+08:00",
```

```
 "type": "string"
 }, {
 "fieldName": "log_ver",
 "content": "elb_01",
 "type": "string"
 }, {
 "fieldName": "remote_addr",
 "content": "192.0.0.0",
 "type": "string"
 }, {
 "fieldName": "remote_port",
 "content": "88888",
 "type": "long"
 }, {
 "fieldName": "status",
 "content": "200",
 "type": "long"
 }, {
 "fieldName": "request_method",
 "content": "GET",
 "type": "string"
 }, {
 "fieldName": "scheme",
 "content": "http",
 "type": "string"
 }, {
 "fieldName": "host",
 "content": "prod.sss.ads.sg2.aaa",
 "type": "string"
 }, {
 "fieldName": "router_request_uri",
 "content": "loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada",
 "type": "string"
 }, {
 "fieldName": "server_protocol",
 "content": "HTTP/1.1",
 "type": "string"
 }, {
 "fieldName": "request_length",
 "content": "233",
 "type": "long"
 }, {
 "fieldName": "bytes_sent",
 "content": "293",
 "type": "long"
 }, {
 "fieldName": "body_bytes_sent",
 "content": "138",
 "type": "long"
 }, {
 "fieldName": "request_time",
 "content": "0.001",
 "type": "float"
 }, {
 "fieldName": "upstream_status",
 "content": "200",
 "type": "long"
 }, {
 "fieldName": "upstream_connect_time",
 "content": "0.000",
 "type": "float"
 }, {
 "fieldName": "upstream_header_time",
 "content": "0.001",
 "type": "float"
 }, {
 "fieldName": "upstream_response_time",
```

```
"content": "0.001",
"type": "float"
}, {
 "fieldName": "upstream_addr",
 "content": "100.0.0.0:9999",
 "type": "string"
}, {
 "fieldName": "http_user_agent",
 "content": "lua-resty-http/0.14 (Lua) ngx_lua/10000",
 "type": "string"
}, {
 "fieldName": "http_referer",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "http_x_forwarded_for",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "lb_name",
 "content": "loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa",
 "type": "string"
}, {
 "fieldName": "listener_name",
 "content": "listener_6077809b-913f-466d-a96c-376f08882d5d",
 "type": "string"
}, {
 "fieldName": "listener_id",
 "content": "08cc2b3f68aa4dddd1e6a90ddd1688348a4480",
 "type": "string"
}, {
 "fieldName": "pool_name",
 "content": "pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b",
 "type": "string"
}, {
 "fieldName": "member_name",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "tenant_id",
 "content": "2fb78dsadadq1213das1121dab146ad3cb0",
 "type": "string"
}, {
 "fieldName": "eip_address",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "eip_port",
 "content": "80",
 "type": "long"
}, {
 "fieldName": "upstream_addr_priv",
 "content": "101.0.0.0:10000",
 "type": "string"
}, {
 "fieldName": "certificate_id",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "ssl_protocol",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "ssl_cipher",
 "content": "-",
 "type": "string"
}, {
 "fieldName": "sni_domain_name",
 "content": "-",

```

```
 "type": "string"
 }, {
 "fieldName": "tcpinfo_rtt",
 "content": "9739",
 "type": "long"
 }],
 "rule": {
 "type": "built_in",
 "param": "ELB"
 },
 "tag_fields": [{
 "fieldName": "hostIP",
 "content": "192.168.2.134",
 "type": "string",
 "isAnalysis": true
 }],
 "log_group_id": " ",
 "log_stream_id": " ",
 "project_id": " ",
 "parse_type": " "
}
```

## 响应示例

### 状态码：200

请求响应成功, 成功修改结构化配置。

```
{
 "2a473356cca5487f8373be891bffc1cf_8a75b77d-7d72-4d7e-8c50-a24562cf8b0b_fd5e1a7c-7412-475d-a013-8891d539574e"
}
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode": "LTS.0612",
 "errorMessage": "timee fieldType is error"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

### 状态码：403

Forbidden。

请求被拒绝访问。

返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Invalid projectId"
}
```

### 状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0102",
 "error_msg": "Failed to create log group"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 修改VPC方式的结构化配置。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateStructTemplateRequest request = new UpdateStructTemplateRequest();
 LtsStructTemplateInfo body = new LtsStructTemplateInfo();
 Rule rulebody = new Rule();
 rulebody.withType("built_in")
 .withParam("VPC");
 List<TagField> listbodyTagFields = new ArrayList<>();
 listbodyTagFields.add(
 new TagField()
 .withFieldName("hostIP")
 .withType("string")
 .withContent("192.168.2.134")
 .withIsAnalysis(true)
);
 List<StructFieldInfo> listbodyDemoFields = new ArrayList<>();
 listbodyDemoFields.add(
```

```
 new StructFieldInfo()
 .withContent("1")
 .withFieldName("version")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("5f67944957444bd6bb4fe3b367de8f3d")
 .withFieldName("project_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("1d515d18-1b36-47dc-a983-bd6512aed4bd")
 .withFieldName("interface_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("192.168.0.154")
 .withFieldName("srcaddr")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("192.168.3.25")
 .withFieldName("dstaddr")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("38929")
 .withFieldName("srcport")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("53")
 .withFieldName("dstport")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("17")
 .withFieldName("protocol")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("1")
 .withFieldName("packets")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("96")
 .withFieldName("bytes")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("1548752136")
 .withFieldName("start")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("1548752736")
 .withFieldName("end")
);
```

```
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("ACCEPT")
 .withFieldName("action")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("OK")
 .withFieldName("log_status")
 .withType("string")
);
 body.withRule(rulebody);
 body.withProjectId(" ");
 body.withLogStreamId(" ");
 body.withParseType(LtsStructTemplateInfo.ParseTypeEnum.fromValue(" "));
 body.withLogGroupId(" ");
 body.withContent("1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-
bd6512aed4bd 192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK");
 body.withTagFields(listbodyTagFields);
 body.withDemoFields(listbodyDemoFields);
 request.withBody(body);
 try {
 UpdateStructTemplateResponse response = client.updateStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

- 修改ELB方式的结构化配置。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateStructTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
```

```
.withProjectId(projectId)
.withAk(ak)
.withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateStructTemplateRequest request = new UpdateStructTemplateRequest();
LtsStructTemplateInfo body = new LtsStructTemplateInfo();
Rule rulebody = new Rule();
rulebody.withType("built_in")
 .withParam("ELB");
List<TagField> listbodyTagFields = new ArrayList<>();
listbodyTagFields.add(
 new TagField()
 .withFieldName("hostIP")
 .withType("string")
 .withContent("192.168.2.134")
 .withIsAnalysis(true)
);
List<StructFieldInfo> listbodyDemoFields = new ArrayList<>();
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("1594727856.337")
 .withFieldName("msec")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("e7c37d97-e922-457c-bbf3-dsadeqac")
 .withFieldName("access_log_topic_id")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("2020-07-14T19:57:36+08:00")
 .withFieldName("time_iso8601")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("elb_01")
 .withFieldName("log_ver")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("192.0.0.0")
 .withFieldName("remote_addr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("88888")
 .withFieldName("remote_port")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("200")
 .withFieldName("status")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("GET")
 .withFieldName("request_method")
 .withType("string")
);
```

```
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("http")
 .withFieldName("scheme")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("prod.sss.ads.sg2.aaa")
 .withFieldName("host")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada")
 .withFieldName("router_request_uri")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("HTTP/1.1")
 .withFieldName("server_protocol")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("233")
 .withFieldName("request_length")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("293")
 .withFieldName("bytes_sent")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("138")
 .withFieldName("body_bytes_sent")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("0.001")
 .withFieldName("request_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("200")
 .withFieldName("upstream_status")
 .withType("long")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("0.000")
 .withFieldName("upstream_connect_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("0.001")
 .withFieldName("upstream_header_time")
 .withType("float")
);
);
```

```
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("0.001")
 .withFieldName("upstream_response_time")
 .withType("float")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("100.0.0.0:9999")
 .withFieldName("upstream_addr")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("lua-resty-http/0.14 (Lua) ngx_lua/10000")
 .withFieldName("http_user_agent")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("http_referer")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("http_x_forwarded_for")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa")
 .withFieldName("lb_name")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("listener_6077809b-913f-466d-a96c-376f08882d5d")
 .withFieldName("listener_name")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("08cc2b3f68aa4ddd1e6a90ddd1688348a4480")
 .withFieldName("listener_id")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b")
 .withFieldName("pool_name")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("member_name")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("2fb78dsadadq1213das1121dab146ad3cb0")
 .withFieldName("tenant_id")
 .withType("string")
);
listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
);
```

```
 .withFieldName("eip_address")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("80")
 .withFieldName("eip_port")
 .withType("long")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("101.0.0.0:10000")
 .withFieldName("upstream_addr_priv")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("certificate_id")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("ssl_protocol")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("ssl_cipher")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("-")
 .withFieldName("sni_domain_name")
 .withType("string")
);
 listbodyDemoFields.add(
 new StructFieldInfo()
 .withContent("9739")
 .withFieldName("tcpinfo_rtt")
 .withType("long")
);
 body.withRule(rulebody);
 body.withProjectId(" ");
 body.withLogStreamId(" ");
 body.withParseType(LtsStructTemplateInfo.ParseTypeEnum.fromValue(" "));
 body.withLogGroupId(" ");
 body.withContent("1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac
2020-07-14T19:57:36+08:00 elb_01 192.0.0.0:88888 200 \"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\" 233 293 138 0.001 \"200\" \"0.000\" \"0.001\" \"0.001\" \"100.0.0.0:9999\"
\"lua-resty-http/0.14 (Lua) ngx_lua/10000\" \"-\" \"-\" loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa
listener_6077809b-913f-466d-a96c-376f08882d5d 08cc2b3f68aa4ddd1e6a90ddd1688348a4480
pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \"-\" 2fb78dsadadq1213das1121dab146ad3cb0 -:80
\"101.0.0.0:10000\" - - - - 9739");
 body.withTagFields(listbodyTagFields);
 body.withDemoFields(listbodyDemoFields);
 request.withBody(body);
 try {
 UpdateStructTemplateResponse response = client.updateStructTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

- 修改VPC方式的结构化配置。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateStructTemplateRequest()
 rulebody = Rule(
 type="built_in",
 param="VPC"
)
 listTagFieldsbody = [
 TagField(
 field_name="hostIP",
 type="string",
 content="192.168.2.134",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 StructFieldInfo(
 content="1",
 field_name="version",
 type="long"
),
 StructFieldInfo(
 content="5f67944957444bd6bb4fe3b367de8f3d",
 field_name="project_id",
 type="string"
),
 StructFieldInfo(
 content="1d515d18-1b36-47dc-a983-bd6512aed4bd",
 field_name="interface_id",
 type="string"
),
 StructFieldInfo(
```

```
 content="192.168.0.154",
 field_name="srcaddr",
 type="string"
),
 StructFieldInfo(
 content="192.168.3.25",
 field_name="dstaddr",
 type="string"
),
 StructFieldInfo(
 content="38929",
 field_name="srcport",
 type="long"
),
 StructFieldInfo(
 content="53",
 field_name="dstport",
 type="long"
),
 StructFieldInfo(
 content="17",
 field_name="protocol",
 type="long"
),
 StructFieldInfo(
 content="1",
 field_name="packets",
 type="long"
),
 StructFieldInfo(
 content="96",
 field_name="bytes",
 type="long"
),
 StructFieldInfo(
 content="1548752136",
 field_name="start",
 type="long"
),
 StructFieldInfo(
 content="1548752736",
 field_name="end",
 type="long"
),
 StructFieldInfo(
 content="ACCEPT",
 field_name="action",
 type="string"
),
 StructFieldInfo(
 content="OK",
 field_name="log_status",
 type="string"
)
]
request.body = LtsStructTemplateInfo(
 rule=rulebody,
 project_id=" ",
 log_stream_id=" ",
 parse_type=" ",
 log_group_id=" ",
 content="1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd
192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK",
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody
)
response = client.update_struct_template(request)
print(response)
except exceptions.ClientRequestException as e:
```

```
print(e.status_code)
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

- 修改ELB方式的结构化配置。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateStructTemplateRequest()
 rulebody = Rule(
 type="built_in",
 param="ELB"
)
 listTagFieldsbody = [
 TagField(
 field_name="hostIP",
 type="string",
 content="192.168.2.134",
 is_analysis=True
)
]
 listDemoFieldsbody = [
 StructFieldInfo(
 content="1594727856.337",
 field_name="msec",
 type="float"
),
 StructFieldInfo(
 content="e7c37d97-e922-457c-bbf3-dsadeqac",
 field_name="access_log_topic_id",
 type="string"
),
 StructFieldInfo(
 content="2020-07-14T19:57:36+08:00",
 field_name="time_iso8601",
 type="string"
),
 StructFieldInfo(
 content="elb_01",
 field_name="log_ver",
 type="string"
),
 StructFieldInfo(
 content="192.0.0.0",
 field_name="remote_addr",
```

```
 type="string"
),
 StructFieldInfo(
 content="88888",
 field_name="remote_port",
 type="long"
),
 StructFieldInfo(
 content="200",
 field_name="status",
 type="long"
),
 StructFieldInfo(
 content="GET",
 field_name="request_method",
 type="string"
),
 StructFieldInfo(
 content="http",
 field_name="scheme",
 type="string"
),
 StructFieldInfo(
 content="prod.sss.ads.sg2.aaa",
 field_name="host",
 type="string"
),
 StructFieldInfo(
 content="location?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada",
 field_name="router_request_uri",
 type="string"
),
 StructFieldInfo(
 content="HTTP/1.1",
 field_name="server_protocol",
 type="string"
),
 StructFieldInfo(
 content="233",
 field_name="request_length",
 type="long"
),
 StructFieldInfo(
 content="293",
 field_name="bytes_sent",
 type="long"
),
 StructFieldInfo(
 content="138",
 field_name="body_bytes_sent",
 type="long"
),
 StructFieldInfo(
 content="0.001",
 field_name="request_time",
 type="float"
),
 StructFieldInfo(
 content="200",
 field_name="upstream_status",
 type="long"
),
 StructFieldInfo(
 content="0.000",
 field_name="upstream_connect_time",
 type="float"
),
),
```

```
StructFieldInfo(
 content="0.001",
 field_name="upstream_header_time",
 type="float"
),
StructFieldInfo(
 content="0.001",
 field_name="upstream_response_time",
 type="float"
),
StructFieldInfo(
 content="100.0.0.0:9999",
 field_name="upstream_addr",
 type="string"
),
StructFieldInfo(
 content="lua-resty-http/0.14 (Lua) ngx_lua/10000",
 field_name="http_user_agent",
 type="string"
),
StructFieldInfo(
 content="-",
 field_name="http_referer",
 type="string"
),
StructFieldInfo(
 content="-",
 field_name="http_x_forwarded_for",
 type="string"
),
StructFieldInfo(
 content="loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa",
 field_name="lb_name",
 type="string"
),
StructFieldInfo(
 content="listener_6077809b-913f-466d-a96c-376f08882d5d",
 field_name="listener_name",
 type="string"
),
StructFieldInfo(
 content="08cc2b3f68aa4ddd1e6a90ddd1688348a4480",
 field_name="listener_id",
 type="string"
),
StructFieldInfo(
 content="pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b",
 field_name="pool_name",
 type="string"
),
StructFieldInfo(
 content="-",
 field_name="member_name",
 type="string"
),
StructFieldInfo(
 content="2fb78dsadadq1213das1121dab146ad3cb0",
 field_name="tenant_id",
 type="string"
),
StructFieldInfo(
 content="-",
 field_name="eip_address",
 type="string"
),
StructFieldInfo(
 content="80",
 field_name="eip_port",
 type="long"
```

```
),
 StructFieldInfo(
 content="101.0.0.0:10000",
 field_name="upstream_addr_priv",
 type="string"
),
 StructFieldInfo(
 content="-",
 field_name="certificate_id",
 type="string"
),
 StructFieldInfo(
 content="-",
 field_name="ssl_protocol",
 type="string"
),
 StructFieldInfo(
 content="-",
 field_name="ssl_cipher",
 type="string"
),
 StructFieldInfo(
 content="-",
 field_name="sni_domain_name",
 type="string"
),
 StructFieldInfo(
 content="9739",
 field_name="tcpinfo_rtt",
 type="long"
)
]
request.body = LtsStructTemplateInfo(
 rule=rulebody,
 project_id=" ",
 log_stream_id=" ",
 parse_type=" ",
 log_group_id=" ",
 content="1594727856.337 e7c37d97-e922-457c-bbf3-dsadeqac 2020-07-14T19:57:36+08:00
elb_01 192.0.0.0:88888 200 \"GET http://prod.sss.ads.sg2.aaa/loc/ation?
version=3&ip=100.0.0.0&coordinate=27.7044784,85.3007481&device_id=dsadsadasdsadasd&beyla_id=w
qeb123ndadsa233ddada HTTP/1.1\" 233 293 138 0.001 \"200\" \"0.000\" \"0.001\" \"0.001\" \"100.0.0.0:9999\"
\"lua-resty-http/0.14 (Lua) ngx_lua/10000\" \"-\" \"-\" loadbalancer_edsaee-4c9c-b467-5b8126b2f7f7dsa
listener_6077809b-913f-466d-a96c-376f08882d5d 08cc2b3f68aa4dddd1e6a90dddd1688348a4480
pool_b2f2966c-043d-4674-ad4b-c15f2adb2c6b \"-\" 2fb78dsadadq1213das1121dab146ad3cb0 -:80
\"101.0.0.0:10000\" - - - - 9739",
 tag_fields=listTagFieldsbody,
 demo_fields=listDemoFieldsbody
)
response = client.update_struct_template(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 修改VPC方式的结构化配置。
- 修改ELB方式的结构化配置。

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                                          |
|-----|-------------------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功修改结构化配置。                                                                                          |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。                                                              |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                             |
| 403 | Forbidden。<br>请求被拒绝访问。<br>返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                             |
| 503 | ServiceUnavailable。<br>被请求的服务无效, 服务不可用。                                                                     |

## 错误码

请参见[错误码](#)。

## 6.9 AOM 容器日志接入 LTS

### 6.9.1 创建接入规则

#### 功能介绍

该接口用于创建AOM容器日志接入LTS的接入规则。

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v2/{project\_id}/lts/aom-mapping

表 6-442 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

表 6-443 Query 参数

| 参数      | 是否必选 | 参数类型    | 描述                                      |
|---------|------|---------|-----------------------------------------|
| isBatch | 是    | Boolean | 是否使用自动映射, 填写为true的话, 可以不用填写容器名、日志流相关参数。 |

## 请求参数

表 6-444 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

表 6-445 请求 Body 参数

| 参数        | 是否必选 | 参数类型                                      | 描述                             |
|-----------|------|-------------------------------------------|--------------------------------|
| rule_name | 是    | String                                    | 接入规则名称<br>最小长度: 1<br>最大长度: 100 |
| rule_info | 是    | <a href="#">AomMappingRuleInfo</a> object | 规则详情                           |

| 参数         | 是否必选 | 参数类型   | 描述                                                                        |
|------------|------|--------|---------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见：<br>10.3-获取账号租户ID、项目资源集ID、日志组ID、日志流ID。<br>最小长度：32<br>最大长度：32 |
| rule_id    | 否    | String | 接入规则id                                                                    |

表 6-446 AomMappingRuleInfo

| 参数                 | 是否必选 | 参数类型                                                       | 描述                                                                         |
|--------------------|------|------------------------------------------------------------|----------------------------------------------------------------------------|
| cluster_id         | 是    | String                                                     | 集群id                                                                       |
| cluster_name       | 是    | String                                                     | 集群名称。                                                                      |
| deployments_prefix | 否    | String                                                     | 日志流前缀。当参数 isBatch=true且填写此参数时，则与参数“deployments”拼接为日志流名称，拼接后的日志流名称长度不超过64位。 |
| deployments        | 是    | Array of strings                                           | 工作负载（选全部工作负载使用“ALL_DEPLOYMENTS”）。<br><b>说明</b><br>自动映射时需要罗列出每个工作负载。        |
| namespace          | 是    | String                                                     | 命名空间                                                                       |
| files              | 是    | Array of <a href="#">CreateAomMappingfilesInfo</a> objects | 日志详细（全部日志使用“ALL_FILES”）。                                                   |
| container_name     | 否    | String                                                     | 容器名称                                                                       |

表 6-447 CreateAomMappingfilesInfo

| 参数        | 是否必选 | 参数类型   | 描述                                                     |
|-----------|------|--------|--------------------------------------------------------|
| file_name | 是    | String | 路径名称。<br>匹配规则：^[A-Za-z0-9.*_/-]+ stdout.log <br>最多两个** |

| 参数              | 是否必选 | 参数类型                                 | 描述        |
|-----------------|------|--------------------------------------|-----------|
| log_stream_info | 是    | CreateAomMappingLogStreamInfo object | 日志流信息日志详细 |

表 6-448 CreateAomMappingLogStreamInfo

| 参数                     | 是否必选 | 参数类型   | 描述                           |
|------------------------|------|--------|------------------------------|
| target_log_group_id    | 是    | String | 日志组ID。<br>最小长度：36<br>最大长度：36 |
| target_log_group_name  | 是    | String | 日志组名称。<br>最小长度：1<br>最大长度：64  |
| target_log_stream_id   | 是    | String | 日志流ID。<br>最小长度：36<br>最大长度：36 |
| target_log_stream_name | 是    | String | 日志流名称。<br>最小长度：1<br>最大长度：64  |

## 响应参数

状态码：201

表 6-449 响应 Body 参数

| 参数     | 参数类型                                      | 描述 |
|--------|-------------------------------------------|----|
| [数组元素] | Array of CreateAomMappingRuleResp objects |    |

表 6-450 CreateAomMappingRuleResp

| 参数         | 参数类型   | 描述   |
|------------|--------|------|
| project_id | String | 项目id |

| 参数        | 参数类型                                              | 描述     |
|-----------|---------------------------------------------------|--------|
| rule_name | String                                            | 接入规则名称 |
| rule_id   | String                                            | 接入规则id |
| rule_info | <a href="#">AomMappingRuleInfoRespBody</a> object | 接入规则详情 |

表 6-451 AomMappingRuleInfoRespBody

| 参数                 | 参数类型                                                 | 描述     |
|--------------------|------------------------------------------------------|--------|
| cluster_id         | String                                               | 集群id   |
| cluster_name       | String                                               | 集群名称   |
| deployments_prefix | String                                               | 日志流前缀  |
| deployments        | Array of strings                                     | 工作负载   |
| namespace          | String                                               | 命名空间   |
| container_name     | String                                               | 容器名称   |
| files              | Array of <a href="#">AomMappingfileInfos</a> objects | 接入规则详情 |

表 6-452 AomMappingfilesInfos

| 参数              | 参数类型                                            | 描述      |
|-----------------|-------------------------------------------------|---------|
| file_name       | String                                          | 路径名     |
| log_stream_info | <a href="#">AomMappingLogStreamInfos</a> object | 接入规则详情。 |

表 6-453 AomMappingLogStreamInfos

| 参数                    | 参数类型   | 描述      |
|-----------------------|--------|---------|
| target_log_group_id   | String | 日志组id   |
| target_log_group_name | String | 目标日志组名称 |

| 参数                     | 参数类型   | 描述      |
|------------------------|--------|---------|
| target_log_stream_id   | String | 日志流id   |
| target_log_stream_name | String | 目标日志流名称 |

状态码：400

表 6-454 响应 Body 参数

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                    |
|---------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| code    | String | 错误码。<br>枚举值： <ul style="list-style-type: none"><li>• LTS.0742</li><li>• LTS.0743</li><li>• LTS.0014</li><li>• LTS.0740</li><li>• LTS.0744</li><li>• LTS.0746</li></ul>                                                                                                                                                                                                                |
| details | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• AOM mapping rule log group id does not exist</li><li>• Operation DB failed</li><li>• AOM mapping rule name already exists</li><li>• AOM mapping rule param validate error</li><li>• AOM mapping rule log stream id does not exist</li><li>• AOM mapping rule log stream name already exist in another log group</li></ul> |

状态码：401

表 6-455 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 状态码：403

表 6-456 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 状态码：500

表 6-457 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

## 请求示例

创建接入规则

POST https://{endpoint}/v2/{project\_id}/lts/aom-mapping

```
{
 "rule_name": "abcde",
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "rule_info": {
 "cluster_id": "4fae3587-0202-11eb-9ba9-0255ac100b02",
 "cluster_name": "testdiskrate",
 "deployments_prefix": "deployments_prefix",
 "deployments": ["__ALL_DEPLOYMENTS__"],
 "namespace": "default",
 "container_name": "container-0",
 "files": [{
 "file_name": "__ALL_FILES__",
 "log_stream_info": {
 "target_log_group_id": "8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 "target_log_group_name": "lts-group-wb28",
 "target_log_stream_id": "2c228bd1-cbf1-41fb-b563-0ca2769202b2",
 "target_log_stream_name": "mysql"
 }
 }
]
}
```

## 响应示例

### 状态码：201

请求响应成功, 成功创建接入配置。

```
[{
 "project_id": "",
 "rule_id": "",
 "rule_info": {
 "cluster_id": "",
 "cluster_name": "",
 "container_name": "",
 "deployments_prefix": "deployments_prefix",
 "deployments": [""],
 "files": [{
 "file_name": "",
 "log_stream_info": {
 "target_log_group_id": "",
 "target_log_group_name": "",
 "target_log_stream_id": "",
 "target_log_stream_name": ""
 }
 }
],
 "namespace": ""
},
"rule_name": ""
]
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode": "LTS.0014",
 "errorMessage": "Operation DB failed"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

**状态码：403**

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code" : "LTS.0003",
 "error_msg" : "parse_token_failed"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0102",
 "error_msg" : "ServiceUnavailable."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 创建接入规则

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateAomMappingRulesSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateAomMappingRulesRequest request = new CreateAomMappingRulesRequest();
```

```
AomMappingRequestInfo body = new AomMappingRequestInfo();
AomMappingLogStreamInfo logStreamInfoFiles = new AomMappingLogStreamInfo();
logStreamInfoFiles.withTargetLogGroupId("8c9dcca6-d048-43a7-989b-c76c34b0ac85")
 .withTargetLogGroupName("lts-group-wb28")
 .withTargetLogStreamId("2c228bd1-cbf1-41fb-b563-0ca2769202b2")
 .withTargetLogStreamName("mysql");
List<AomMappingfilesInfo> listRuleInfoFiles = new ArrayList<>();
listRuleInfoFiles.add(
 new AomMappingfilesInfo()
 .withFileName("__ALL_FILES__")
 .withLogStreamInfo(logStreamInfoFiles)
);
List<String> listRuleInfoDeployments = new ArrayList<>();
listRuleInfoDeployments.add("__ALL_DEPLOYMENTS__");
AomMappingRuleInfo ruleInfobody = new AomMappingRuleInfo();
ruleInfobody.withClusterId("4fae3587-0202-11eb-9ba9-0255ac100b02")
 .withClusterName("testdiskrate")
 .withDeploymentsPrefix("deployments_prefix")
 .withDeployments(listRuleInfoDeployments)
 .withNamespace("default")
 .withContainerName("container-0")
 .withFiles(listRuleInfoFiles);
body.withRuleInfo(ruleInfobody);
body.withRuleName("abcde");
body.withProjectId("2a473356cca5487f8373be891bffc1cf");
request.withBody(body);
try {
 CreateAomMappingRulesResponse response = client.createAomMappingRules(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 创建接入规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
```

```
.with_region(LtsRegion.value_of("<YOUR REGION>")) \
.build()

try:
 request = CreateAomMappingRulesRequest()
 logStreamInfoFiles = AomMappingLogStreamInfo(
 target_log_group_id="8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 target_log_group_name="lts-group-wb28",
 target_log_stream_id="2c228bd1-cbf1-41fb-b563-0ca2769202b2",
 target_log_stream_name="mysql"
)
 listFilesRuleInfo = [
 AomMappingfilesInfo(
 file_name="__ALL_FILES__",
 log_stream_info=logStreamInfoFiles
)
]
 listDeploymentsRuleInfo = [
 "__ALL_DEPLOYMENTS__"
]
 ruleInfobody = AomMappingRuleInfo(
 cluster_id="4fae3587-0202-11eb-9ba9-0255ac100b02",
 cluster_name="testdiskrate",
 deployments_prefix="deployments_prefix",
 deployments=listDeploymentsRuleInfo,
 namespace="default",
 container_name="container-0",
 files=listFilesRuleInfo
)
 request.body = AomMappingRequestInfo(
 rule_info=ruleInfobody,
 rule_name="abcde",
 project_id="2a473356cca5487f8373be891bffc1cf"
)
 response = client.create_aom_mapping_rules(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建接入规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
```

```
WithProjectId(projectId).
Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.CreateAomMappingRulesRequest{}
logStreamInfoFiles := &model.AomMappingLogStreamInfo{
 TargetLogGroupId: "8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 TargetLogGroupName: "lts-group-wb28",
 TargetLogStreamId: "2c228bd1-cbf1-41fb-b563-0ca2769202b2",
 TargetLogStreamName: "mysql",
}
var listFilesRuleInfo = []model.AomMappingfilesInfo{
 {
 FileName: "__ALL_FILES__",
 LogStreamInfo: logStreamInfoFiles,
 },
}
var listDeploymentsRuleInfo = []string{
 "__ALL_DEPLOYMENTS__",
}
deploymentsPrefixRuleInfo:= "deployments_prefix"
containerNameRuleInfo:= "container-0"
ruleInfobody := &model.AomMappingRuleInfo{
 ClusterId: "4fae3587-0202-11eb-9ba9-0255ac100b02",
 ClusterName: "testdiskrate",
 DeploymentsPrefix: &deploymentsPrefixRuleInfo,
 Deployments: listDeploymentsRuleInfo,
 Namespace: "default",
 ContainerName: &containerNameRuleInfo,
 Files: listFilesRuleInfo,
}
request.Body = &model.AomMappingRequestInfo{
 RuleInfo: ruleInfobody,
 RuleName: "abcde",
 ProjectId: "2a473356cca5487f8373be891bffc1cf",
}
response, err := client.CreateAomMappingRules(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                            |
|-----|-----------------------------------------------|
| 201 | 请求响应成功, 成功创建接入配置。                             |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。 |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。               |

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到，但是服务内部出错。                                                |
| 503 | ServiceUnavailable。<br>被请求的服务无效,服务不可用。                                                        |

## 错误码

请参见[错误码](#)。

## 6.9.2 修改接入规则

### 功能介绍

该接口用于修改接入规则

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v2/{project\_id}/lts/aom-mapping

表 6-458 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                                                                  |
|------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-459 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-460 请求 Body 参数

| 参数         | 是否必选 | 参数类型                                            | 描述                           |
|------------|------|-------------------------------------------------|------------------------------|
| rule_id    | 是    | String                                          | 接入规则id<br>最小长度：36<br>最大长度：36 |
| project_id | 是    | String                                          | 项目id<br>最小长度：32<br>最大长度：32   |
| rule_name  | 是    | String                                          | 接入规则名称<br>最小长度：1<br>最大长度：100 |
| rule_info  | 是    | <a href="#">UpdateAomMappingRuleInfo</a> object | 接入规则详情                       |

表 6-461 UpdateAomMappingRuleInfo

| 参数           | 是否必选 | 参数类型   | 描述    |
|--------------|------|--------|-------|
| cluster_id   | 是    | String | 集群ID。 |
| cluster_name | 是    | String | 集群名称。 |

| 参数                 | 是否必选 | 参数类型                                                 | 描述                                                                  |
|--------------------|------|------------------------------------------------------|---------------------------------------------------------------------|
| deployments        | 是    | Array of strings                                     | 工作负载（选全部工作负载使用“ALL_DEPLOYMENTS”）。<br><b>说明</b><br>自动映射时需要罗列出每个工作负载。 |
| namespace          | 是    | String                                               | 命名空间。                                                               |
| files              | 是    | Array of <a href="#">AomMappingfilesInfo</a> objects | 日志详细（全部日志使用“ALL_FILES”）。                                            |
| container_name     | 否    | String                                               | 容器名称                                                                |
| deployments_prefix | 否    | String                                               | 可选择的自定义日志流前缀                                                        |

表 6-462 AomMappingfilesInfo

| 参数              | 是否必选 | 参数类型                                                 | 描述                                              |
|-----------------|------|------------------------------------------------------|-------------------------------------------------|
| file_name       | 是    | String                                               | 路径名,匹配规则: ^/[A-Za-z0-9.*_/-]+ stdout.log 最多两个** |
| log_stream_info | 是    | <a href="#">UpdateAomMappingLogStreamInfo</a> object | 日志流信息日志详细。                                      |

表 6-463 UpdateAomMappingLogStreamInfo

| 参数                    | 是否必选 | 参数类型   | 描述                                                          |
|-----------------------|------|--------|-------------------------------------------------------------|
| target_log_group_id   | 是    | String | 日志组ID。日志组ID不为空时，必须有效。<br>最小长度： <b>36</b><br>最大长度： <b>36</b> |
| target_log_group_name | 是    | String | 日志组名称。日志组名称与ID不能同时为空。<br>最小长度： <b>1</b><br>最大长度： <b>64</b>  |

| 参数                     | 是否必选 | 参数类型   | 描述                                          |
|------------------------|------|--------|---------------------------------------------|
| target_log_stream_id   | 是    | String | 日志流ID。日志流ID不为空时，必须有效。<br>最小长度：36<br>最大长度：36 |
| target_log_stream_name | 是    | String | 日志流名称。日志流名称与ID不能同时为空。<br>最小长度：1<br>最大长度：64  |

## 响应参数

状态码：200

表 6-464 响应 Body 参数

| 参数         | 参数类型                                                    | 描述     |
|------------|---------------------------------------------------------|--------|
| project_id | String                                                  | 项目id   |
| rule_name  | String                                                  | 接入规则名称 |
| rule_id    | String                                                  | 接入规则id |
| rule_info  | <a href="#">UpdateAomMappingRuleInfoRespBody</a> object | 接入规则详情 |

表 6-465 UpdateAomMappingRuleInfoRespBody

| 参数             | 参数类型                                                        | 描述                                                           |
|----------------|-------------------------------------------------------------|--------------------------------------------------------------|
| cluster_id     | String                                                      | 集群ID。                                                        |
| cluster_name   | String                                                      | 集群名称。                                                        |
| deployments    | Array of strings                                            | 工作负载（选全部工作负载使用“ALL_DEPLOYMENTS”）。<br>说明<br>自动映射时需要罗列出每个工作负载。 |
| namespace      | String                                                      | 命名空间。                                                        |
| files          | Array of <a href="#">UpdateAomMappingfilesInfos</a> objects | 日志详细（全部日志使用“ALL_FILES”）。                                     |
| container_name | String                                                      | 容器名称                                                         |

| 参数                     | 参数类型   | 描述           |
|------------------------|--------|--------------|
| deployments_prefi<br>x | String | 可选择的自定义日志流前缀 |

表 6-466 UpdateAomMappingfilesInfos

| 参数              | 参数类型                                                   | 描述                                                           |
|-----------------|--------------------------------------------------------|--------------------------------------------------------------|
| file_name       | String                                                 | 路径名称。<br>匹配规则: ^/[A-Za-z0-9.*_/-]+ <br>stdout.log <br>最多两个** |
| log_stream_info | <b>UpdateAomMap<br/>pingLogStreamIn<br/>fos</b> object | 日志流信息日志详细。                                                   |

表 6-467 UpdateAomMappingLogStreamInfos

| 参数                         | 参数类型   | 描述                                                                        |
|----------------------------|--------|---------------------------------------------------------------------------|
| target_log_group_<br>id    | String | 日志组ID。日志组ID不为空时, 必须有效。<br>缺省值: None<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| target_log_group_<br>name  | String | 日志组名称。日志组名称与ID不能同时<br>为空。<br>最小长度: <b>1</b><br>最大长度: <b>64</b>            |
| target_log_stream_<br>id   | String | 日志流ID。日志流ID不为空时, 必须有效。<br>缺省值: None<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| target_log_stream_<br>name | String | 日志流名称。日志流名称与ID不能同时<br>为空。<br>最小长度: <b>1</b><br>最大长度: <b>64</b>            |

状态码: 400

表 6-468 响应 Body 参数

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                           |
|---------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| code    | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• LTS.0742</li> <li>• LTS.0743</li> <li>• LTS.0014</li> <li>• LTS.0740</li> <li>• LTS.0744</li> <li>• LTS.0746</li> </ul>                                                                                                                                                                                                                |
| details | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• AOM mapping rule log group id does not exist</li> <li>• Operation DB failed</li> <li>• AOM mapping rule name already exists</li> <li>• AOM mapping rule param validate error</li> <li>• AOM mapping rule log stream id does not exist</li> <li>• AOM mapping rule log stream name already exist in another log group</li> </ul> |

状态码：401

表 6-469 响应 Body 参数

| 参数      | 参数类型                        | 描述      |
|---------|-----------------------------|---------|
| message | <b>Message401</b><br>object | 接口调用信息。 |

表 6-470 Message401

| 参数   | 参数类型   | 描述                                                                        |
|------|--------|---------------------------------------------------------------------------|
| code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>• LTS.0001</li> </ul> |

| 参数      | 参数类型   | 描述                                                                                                |
|---------|--------|---------------------------------------------------------------------------------------------------|
| details | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• <b>project verify error</b></li></ul> |

状态码：403

表 6-471 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                             |
|------------|--------|------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"><li>• <b>LTS.0403</b></li></ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• <b>Invalid projectId</b></li></ul> |

状态码：500

表 6-472 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                             |
|------------|--------|------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"><li>• <b>LTS.0403</b></li></ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• <b>Invalid projectId</b></li></ul> |

## 请求示例

修改接入规则

```
PUT https://{endpoint}/v2/{project_id}/lts/aom-mapping
```

```
{
 "rule_id": "",
 "rule_name": "abcde",
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "rule_info": {
 "cluster_id": "4fae3587-0202-11eb-9ba9-0255ac100b02",
 "cluster_name": "testdiskrate",
```

```
"deployments": ["__ALL_DEPLOYMENTS__"],
"namespace": "default",
"container_name": "container-0",
"files": [{
 "file_name": "__ALL_FILES__",
 "log_stream_info": {
 "target_log_group_id": "8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 "target_log_group_name": "lts-group-wb28",
 "target_log_stream_id": "2c228bd1-cbf1-41fb-b563-0ca2769202b2",
 "target_log_stream_name": "mysql"
 }
}
]
}
```

## 响应示例

### 状态码：200

请求响应成功, 成功更新接入配置

```
{
 "rule_id": "",
 "project_id": "",
 "rule_name": "",
 "rule_info": {
 "cluster_name": "",
 "cluster_id": "",
 "deployments": [],
 "container_name": "",
 "namespace": "",
 "files": [{
 "file_name": "",
 "log_stream_info": {
 "target_log_stream_name": "",
 "target_log_group_name": "",
 "target_log_stream_id": "",
 "target_log_group_id": ""
 }
 }
]
}
```

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode": "LTS.0014",
 "errorMessage": "Operation DB failed"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0003",
```

```
"error_msg" : "parse_token_failed"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0102",
 "error_msg" : "ServiceUnavailable."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 修改接入规则

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateAomMappingRulesSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateAomMappingRulesRequest request = new UpdateAomMappingRulesRequest();
 UpdateAomMappingRequest body = new UpdateAomMappingRequest();
 AomMappingLogStreamInfo logStreamInfoFiles = new AomMappingLogStreamInfo();
 logStreamInfoFiles.withTargetLogGroupId("8c9dcda6-d048-43a7-989b-c76c34b0ac85")
 .withTargetLogGroupName("lts-group-wb28")
 .withTargetLogStreamId("2c228bd1-cbf1-41fb-b563-0ca2769202b2")
 .withTargetLogStreamName("mysql");
 List<AomMappingfilesInfo> listRuleInfoFiles = new ArrayList<>();
 listRuleInfoFiles.add(
 new AomMappingfilesInfo()
```

```
 .withFileName("__ALL_FILES__")
 .withLogStreamInfo(logStreamInfoFiles)
);
 List<String> listRuleInfoDeployments = new ArrayList<>();
 listRuleInfoDeployments.add("__ALL_DEPLOYMENTS__");
 AomMappingRuleInfo ruleInfobody = new AomMappingRuleInfo();
 ruleInfobody.withClusterId("4fae3587-0202-11eb-9ba9-0255ac100b02")
 .withClusterName("testdiskrate")
 .withDeployments(listRuleInfoDeployments)
 .withNamespace("default")
 .withContainerName("container-0")
 .withFiles(listRuleInfoFiles);
 body.withRuleInfo(ruleInfobody);
 body.withRuleName("abcde");
 body.withProjectId("2a473356cca5487f8373be891bffc1cf");
 body.withRuleId("");
 request.withBody(body);
 try {
 UpdateAomMappingRulesResponse response = client.updateAomMappingRules(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 修改接入规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateAomMappingRulesRequest()
 logStreamInfoFiles = AomMappingLogStreamInfo(
 target_log_group_id="8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 target_log_group_name="lts-group-wb28",
 target_log_stream_id="2c228bd1-cbf1-41fb-b563-0ca2769202b2",
```

```
 target_log_stream_name="mysql"
)
 listFilesRuleInfo = [
 AomMappingfilesInfo(
 file_name="__ALL_FILES_",
 log_stream_info=logStreamInfoFiles
)
]
 listDeploymentsRuleInfo = [
 "__ALL_DEPLOYMENTS_"
]
 ruleInfobody = AomMappingRuleInfo(
 cluster_id="4fae3587-0202-11eb-9ba9-0255ac100b02",
 cluster_name="testdiskrate",
 deployments=listDeploymentsRuleInfo,
 namespace="default",
 container_name="container-0",
 files=listFilesRuleInfo
)
 request.body = UpdateAomMappingRequest(
 rule_info=ruleInfobody,
 rule_name="abcde",
 project_id="2a473356cca5487f8373be891bffc1cf",
 rule_id=""
)
 response = client.update_aom_mapping_rules(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 修改接入规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())
```

```
request := &model.UpdateAomMappingRulesRequest{
logStreamInfoFiles := &model.AomMappingLogStreamInfo{
 TargetLogGroupId: "8c9dcda6-d048-43a7-989b-c76c34b0ac85",
 TargetLogGroupName: "lts-group-wb28",
 TargetLogStreamId: "2c228bd1-cbf1-41fb-b563-0ca2769202b2",
 TargetLogStreamName: "mysql",
}
var listFilesRuleInfo = []model.AomMappingfilesInfo{
 {
 FileName: "__ALL_FILES__",
 LogStreamInfo: logStreamInfoFiles,
 },
}
var listDeploymentsRuleInfo = []string{
 "__ALL_DEPLOYMENTS__",
}
containerNameRuleInfo:= "container-0"
ruleInfobody := &model.AomMappingRuleInfo{
 ClusterId: "4fae3587-0202-11eb-9ba9-0255ac100b02",
 ClusterName: "testdiskrate",
 Deployments: listDeploymentsRuleInfo,
 Namespace: "default",
 ContainerName: &containerNameRuleInfo,
 Files: listFilesRuleInfo,
}
request.Body = &model.UpdateAomMappingRequest{
 RuleInfo: ruleInfobody,
 RuleName: "abcde",
 ProjectId: "2a473356cca5487f8373be891bffc1cf",
 RuleId: "",
}
response, err := client.UpdateAomMappingRules(request)
if err == nil {
 fmt.Printf("%v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功更新接入配置                                                                              |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。                                                 |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                               |
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到，但是服务内部出错。                                                |

| 状态码 | 描述                                     |
|-----|----------------------------------------|
| 503 | ServiceUnavailable。<br>被请求的服务无效,服务不可用。 |

## 错误码

请参见[错误码](#)。

## 6.9.3 删除接入规则

### 功能介绍

该接口用于删除AOM容器日志接入LTS的接入规则。

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v2/{project\_id}/lts/aom-mapping

表 6-473 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

表 6-474 Query 参数

| 参数 | 是否必选 | 参数类型   | 描述                              |
|----|------|--------|---------------------------------|
| id | 是    | String | 接入规则ID。<br>最小长度: 36<br>最大长度: 36 |

## 请求参数

表 6-475 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-476 响应 Body 参数

| 参数     | 参数类型             | 描述 |
|--------|------------------|----|
| [数组元素] | Array of strings |    |

状态码：400

表 6-477 响应 Body 参数

| 参数      | 参数类型   | 描述                                                                                                                                                                               |
|---------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| code    | String | 错误码。<br>枚举值： <ul style="list-style-type: none"> <li>LTS.0744</li> <li>LTS.0014</li> <li>LTS.0745</li> </ul>                                                                      |
| details | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>AOM mapping rule param validate error</li> <li>Operation DB failed</li> <li>AOM mapping rule id is invalid</li> </ul> |

**状态码：401****表 6-478** 响应 Body 参数

| 参数      | 参数类型                        | 描述      |
|---------|-----------------------------|---------|
| message | <b>Message401</b><br>object | 接口调用信息。 |

**表 6-479** Message401

| 参数      | 参数类型   | 描述                                                   |
|---------|--------|------------------------------------------------------|
| code    | String | 错误码。<br>枚举值：<br>• <b>LTS.0001</b>                    |
| details | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>project verify error</b> |

**状态码：403****表 6-480** 响应 Body 参数

| 参数         | 参数类型   | 描述                                                |
|------------|--------|---------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Invalid projectId</b> |

**状态码：500****表 6-481** 响应 Body 参数

| 参数         | 参数类型   | 描述                                |
|------------|--------|-----------------------------------|
| error_code | String | 错误码。<br>枚举值：<br>• <b>LTS.0403</b> |

| 参数        | 参数类型   | 描述                                                                                    |
|-----------|--------|---------------------------------------------------------------------------------------|
| error_msg | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

删除接入规则

```
DELETE /v2/{project_id}/lts/aom-mapping?id={id}
```

```
DELETE https://{endpoint}/v2/{project_id}/lts/aom-mapping?id=3e14a212-a633-411d-a8e1-dc4042b1471a
```

## 响应示例

### 状态码：400

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode": "LTS.0014",
 "errorMessage": "Operation DB failed"
}
```

### 状态码：401

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code": "LTS.0414",
 "error_msg": "Invalid token"
}
```

### 状态码：403

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code": "LTS.0003",
 "error_msg": "parse_token_failed"
}
```

### 状态码：500

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.0102",
 "error_msg": "ServiceUnavailable."
}
```

## 状态码

| 状态码 | 描述                                                                                                  |
|-----|-----------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功删除接入配置。                                                                                   |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。                                                      |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                     |
| 403 | Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                     |

## 错误码

请参见[错误码](#)。

## 6.9.4 查询所有接入规则

### 功能介绍

该接口用于查询所有AOM容器日志接入LTS的接入规则。

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/lts/aom-mapping

表 6-482 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

表 6-483 Query 参数

| 参数              | 是否必选 | 参数类型   | 描述                         |
|-----------------|------|--------|----------------------------|
| log_group_name  | 否    | String | 日志组名称<br>最小长度：1<br>最大长度：64 |
| log_stream_name | 否    | String | 日志流名称<br>最小长度：1<br>最大长度：64 |

## 请求参数

表 6-484 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-485 响应 Body 参数

| 参数     | 参数类型                                                      | 描述 |
|--------|-----------------------------------------------------------|----|
| [数组元素] | Array of <a href="#">CreateAomMappingRuleResp</a> objects |    |

表 6-486 CreateAomMappingRuleResp

| 参数         | 参数类型                                              | 描述     |
|------------|---------------------------------------------------|--------|
| project_id | String                                            | 项目id   |
| rule_name  | String                                            | 接入规则名称 |
| rule_id    | String                                            | 接入规则id |
| rule_info  | <a href="#">AomMappingRuleInfoRespBody</a> object | 接入规则详情 |

表 6-487 AomMappingRuleInfoRespBody

| 参数                 | 参数类型                                                 | 描述     |
|--------------------|------------------------------------------------------|--------|
| cluster_id         | String                                               | 集群id   |
| cluster_name       | String                                               | 集群名称   |
| deployments_prefix | String                                               | 日志流前缀  |
| deployments        | Array of strings                                     | 工作负载   |
| namespace          | String                                               | 命名空间   |
| container_name     | String                                               | 容器名称   |
| files              | Array of <a href="#">AomMappingfileInfos</a> objects | 接入规则详情 |

表 6-488 AomMappingfilesInfos

| 参数              | 参数类型                                            | 描述      |
|-----------------|-------------------------------------------------|---------|
| file_name       | String                                          | 路径名     |
| log_stream_info | <a href="#">AomMappingLogStreamInfos</a> object | 接入规则详情。 |

表 6-489 AomMappingLogStreamInfos

| 参数                  | 参数类型   | 描述    |
|---------------------|--------|-------|
| target_log_group_id | String | 日志组id |

| 参数                     | 参数类型   | 描述      |
|------------------------|--------|---------|
| target_log_group_name  | String | 目标日志组名称 |
| target_log_stream_id   | String | 日志流id   |
| target_log_stream_name | String | 目标日志流名称 |

状态码：400

表 6-490 响应 Body 参数

| 参数      | 参数类型   | 描述                                                  |
|---------|--------|-----------------------------------------------------|
| code    | String | 错误码。<br>枚举值：<br>• <b>LTS.0014</b>                   |
| details | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>Operation DB failed</b> |

状态码：401

表 6-491 响应 Body 参数

| 参数      | 参数类型                        | 描述      |
|---------|-----------------------------|---------|
| message | <b>Message401</b><br>object | 接口调用信息。 |

表 6-492 Message401

| 参数      | 参数类型   | 描述                                                   |
|---------|--------|------------------------------------------------------|
| code    | String | 错误码。<br>枚举值：<br>• <b>LTS.0001</b>                    |
| details | String | 调用失败响应信息描述。<br>枚举值：<br>• <b>project verify error</b> |

**状态码：403****表 6-493 响应 Body 参数**

| 参数         | 参数类型   | 描述                                                                                             |
|------------|--------|------------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"><li>• <b>LTS.0403</b></li></ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• <b>Invalid projectId</b></li></ul> |

**状态码：500****表 6-494 响应 Body 参数**

| 参数         | 参数类型   | 描述                                                                             |
|------------|--------|--------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值： <ul style="list-style-type: none"><li>• <b>LTS.0403</b></li></ul> |
| error_msg  | String | 调用失败响应信息描述。                                                                    |

**请求示例**

无

**响应示例****状态码：200**

请求响应成功, 成功获取所有接入配置。

```
[{
 "project_id": "",
 "rule_id": "",
 "rule_info": {
 "cluster_id": "",
 "cluster_name": "",
 "container_name": "",
 "deployments": [""],
 "files": [{
 "file_name": "",
 "log_stream_info": {
 "target_log_group_id": "",
 "target_log_group_name": "",
 "target_log_stream_id": "",
 "target_log_stream_name": ""
 }
 }
]
},
]
```

```
"namespace" : ""
},
"rule_name" : ""
}, {
"project_id" : "",
"rule_id" : "",
"rule_info" : {
"cluster_id" : "",
"cluster_name" : "",
"container_name" : "",
"deployments" : [""],
"files" : [{
"file_name" : "",
"log_stream_info" : {
"target_log_group_id" : "",
"target_log_group_name" : "",
"target_log_stream_id" : "",
"target_log_stream_name" : ""
}
}
}],
"namespace" : ""
},
"rule_name" : ""
}]
```

**状态码：400**

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
"errorCode" : "LTS.0014",
"errorMessage" : "Operation DB failed"
}
```

**状态码：401**

AuthFailed。鉴权失败，请确认token后再次请求。

```
{
"error_code" : "LTS.0414",
"error_msg" : "Invalid token"
}
```

**状态码：403**

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
"error_code" : "LTS.0003",
"error_msg" : "parse_token_failed"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
"error_code" : "LTS.0102",
"error_msg" : "ServiceUnavailable."
}
```

## SDK 代码示例

SDK代码示例如下。

## Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ShowAomMappingRulesSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ShowAomMappingRulesRequest request = new ShowAomMappingRulesRequest();
 try {
 ShowAomMappingRulesResponse response = client.showAomMappingRules(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

try:
 request = ShowAomMappingRulesRequest()
 response = client.show_aom_mapping_rules(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ShowAomMappingRulesRequest{}
 response, err := client.ShowAomMappingRules(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                                                                                  |
|-----|-----------------------------------------------------------------------------------------------------|
| 200 | 请求响应成功, 成功获取所有接入配置。                                                                                 |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求, 不要重试该请求。                                                      |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。                                                                     |
| 403 | Forbidden。请求被拒绝访问。返回该状态码, 表明请求能够到达服务端, 且服务端能够理解用户请求, 但是拒绝做更多的事情, 因为该请求被设置为拒绝访问, 建议直接修改该请求, 不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到, 但是服务内部出错。                                                     |
| 503 | ServiceUnavailable。<br>被请求的服务无效, 服务不可用。                                                             |

## 错误码

请参见[错误码](#)。

### 6.9.5 查询单个接入规则

#### 功能介绍

该接口用于查询单个AOM容器日志接入LTS的接入规则。

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v2/{project\_id}/lts/aom-mapping/{rule\_id}

表 6-495 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

| 参数      | 是否必选 | 参数类型   | 描述                            |
|---------|------|--------|-------------------------------|
| rule_id | 是    | String | 接入规则ID。<br>最小长度：36<br>最大长度：36 |

## 请求参数

表 6-496 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-497 响应 Body 参数

| 参数     | 参数类型                                                      | 描述 |
|--------|-----------------------------------------------------------|----|
| [数组元素] | Array of <a href="#">CreateAomMappingRuleResp</a> objects |    |

表 6-498 CreateAomMappingRuleResp

| 参数         | 参数类型   | 描述     |
|------------|--------|--------|
| project_id | String | 项目id   |
| rule_name  | String | 接入规则名称 |
| rule_id    | String | 接入规则id |

| 参数        | 参数类型                                              | 描述     |
|-----------|---------------------------------------------------|--------|
| rule_info | <a href="#">AomMappingRuleInfoRespBody</a> object | 接入规则详情 |

表 6-499 AomMappingRuleInfoRespBody

| 参数                 | 参数类型                                                 | 描述     |
|--------------------|------------------------------------------------------|--------|
| cluster_id         | String                                               | 集群id   |
| cluster_name       | String                                               | 集群名称   |
| deployments_prefix | String                                               | 日志流前缀  |
| deployments        | Array of strings                                     | 工作负载   |
| namespace          | String                                               | 命名空间   |
| container_name     | String                                               | 容器名称   |
| files              | Array of <a href="#">AomMappingfileInfos</a> objects | 接入规则详情 |

表 6-500 AomMappingfilesInfos

| 参数              | 参数类型                                            | 描述      |
|-----------------|-------------------------------------------------|---------|
| file_name       | String                                          | 路径名     |
| log_stream_info | <a href="#">AomMappingLogStreamInfos</a> object | 接入规则详情。 |

表 6-501 AomMappingLogStreamInfos

| 参数                    | 参数类型   | 描述      |
|-----------------------|--------|---------|
| target_log_group_id   | String | 日志组id   |
| target_log_group_name | String | 目标日志组名称 |
| target_log_stream_id  | String | 日志流id   |

| 参数                     | 参数类型   | 描述      |
|------------------------|--------|---------|
| target_log_stream_name | String | 目标日志流名称 |

状态码：400

表 6-502 响应 Body 参数

| 参数      | 参数类型   | 描述                                                                                             |
|---------|--------|------------------------------------------------------------------------------------------------|
| code    | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>• LTS.0014</li> </ul>                   |
| details | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>• Operation DB failed</li> </ul> |

状态码：401

表 6-503 响应 Body 参数

| 参数      | 参数类型                        | 描述      |
|---------|-----------------------------|---------|
| message | <b>Message401</b><br>object | 接口调用信息。 |

表 6-504 Message401

| 参数      | 参数类型   | 描述                                                                                              |
|---------|--------|-------------------------------------------------------------------------------------------------|
| code    | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>• LTS.0001</li> </ul>                    |
| details | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>• project verify error</li> </ul> |

状态码：403

表 6-505 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

状态码：500

表 6-506 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。<br>枚举值：<br><ul style="list-style-type: none"> <li>LTS.0403</li> </ul>                 |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

## 请求示例

无

## 响应示例

状态码：200

请求响应成功, 成功获取单个接入配置。

```
[{
 "project_id": "",
 "rule_id": "",
 "rule_info": {
 "cluster_id": "",
 "cluster_name": "",
 "container_name": "",
 "deployments": [""],
 "files": [{
 "file_name": "",
 "log_stream_info": {
 "target_log_group_id": "",
 "target_log_group_name": "",
 "target_log_stream_id": "",
 "target_log_stream_name": ""
 }
 }
]
}]
```

```
"namespace" : ""
},
"rule_name" : ""
}]
```

**状态码：400**

BadRequest。非法请求。建议根据error\_msg直接修改该请求，不要重试该请求。

```
{
 "errorCode" : "LTS.0014",
 "errorMessage" : "Operation DB failed"
}
```

**状态码：401**

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
 "error_code" : "LTS.0414",
 "error_msg" : "Invalid token"
}
```

**状态码：403**

Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。

```
{
 "error_code" : "LTS.0003",
 "error_msg" : "parse_token_failed"
}
```

**状态码：500**

InternalServerError。

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.0102",
 "error_msg" : "ServiceUnavailable."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ShowAomMappingRuleSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD\_SDK\_AK and CLOUD\_SDK\_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
ShowAomMappingRuleRequest request = new ShowAomMappingRuleRequest();
request.withRuleId("{rule_id}");
try {
 ShowAomMappingRuleResponse response = client.showAomMappingRule(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ShowAomMappingRuleRequest()
 request.rule_id = "{rule_id}"
 response = client.show_aom_mapping_rule(request)
 print(response)
 except exceptions.ClientRequestException as e:
```

```
print(e.status_code)
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ShowAomMappingRuleRequest{}
 request.RuleId = "{rule_id}"
 response, err := client.ShowAomMappingRule(request)
 if err == nil {
 fmt.Printf("%v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                            |
|-----|-----------------------------------------------|
| 200 | 请求响应成功, 成功获取单个接入配置。                           |
| 400 | BadRequest。非法请求。建议根据error_msg直接修改该请求，不要重试该请求。 |
| 401 | AuthFailed。鉴权失败, 请确认token后再次请求。               |

| 状态码 | 描述                                                                                            |
|-----|-----------------------------------------------------------------------------------------------|
| 403 | Forbidden。请求被拒绝访问。返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。 |
| 500 | InternalServerError。<br>表明服务端能被请求访问到，但是服务内部出错。                                                |
| 503 | ServiceUnavailable。<br>被请求的服务无效，服务不可用。                                                        |

## 错误码

请参见[错误码](#)。

## 6.10 告警主题

### 6.10.1 查询 SMN 主题

#### 功能介绍

该接口用于查询SMN主题。

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v2/{project\_id}/lts/notifications/topics

表 6-507 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-508 Query 参数

| 参数         | 是否必选 | 参数类型    | 描述                                             |
|------------|------|---------|------------------------------------------------|
| offset     | 是    | Integer | 查询游标，初始传入0，后续从上一次的返回值中获取。<br>最小值：0<br>最大值：1024 |
| limit      | 是    | Integer | 每页数据量，最大值为100。<br>最小值：0<br>最大值：100             |
| fuzzy_name | 否    | String  | 检索的主题名称，模糊匹配，按照startwith模式进行匹配。                |

## 请求参数

表 6-509 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-510 响应 Body 参数

| 参数         | 参数类型   | 描述                              |
|------------|--------|---------------------------------|
| request_id | String | request_id<br>最小长度：1<br>最大长度：64 |

| 参数          | 参数类型                           | 描述                            |
|-------------|--------------------------------|-------------------------------|
| topic_count | Integer                        | topics数量<br>最小值：0<br>最大值：1000 |
| topics      | Array of <b>Topics</b> objects | 主题信息                          |

表 6-511 Topics

| 参数           | 参数类型    | 描述                                                  |
|--------------|---------|-----------------------------------------------------|
| name         | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

## 请求示例

### 查询SMN主题

```
POST https://{endpoint}/v2/{project_id}/lts/notifications/topics
/v2/{project_id}/lts/notifications/topics?offset={offset}&limit={limit}
```

## 响应示例

### 状态码：200

请求响应成功。

```
{
 "request_id": "1",
 "topic_count": 100,
 "topics": [
 {
 "name": "huawei",
 "topic_urn": "urn:smn:cn-north-7:{projectId}:fyy",
 "display_name": "",
```

```
"push_policy": 0
}]
}
```

## 状态码

| 状态码 | 描述      |
|-----|---------|
| 200 | 请求响应成功。 |

## 错误码

请参见[错误码](#)。

# 6.11 消息模板管理

## 6.11.1 创建消息模板

### 功能介绍

该接口用于创建通知模板，目前每个账户最多可以创建共100个通知模板，创建后名称不可修改。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/{domain\_id}/lts/events/notification/templates

表 6-512 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id  | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-513 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                            |
|--------------|------|--------|-----------------------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度： <b>1000</b><br>最大长度： <b>2000</b> |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度： <b>30</b><br>最大长度： <b>30</b>                   |

表 6-514 请求 Body 参数

| 参数     | 是否必选 | 参数类型             | 描述                                                                                                       |
|--------|------|------------------|----------------------------------------------------------------------------------------------------------|
| name   | 是    | String           | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度： <b>1</b><br>最大长度： <b>100</b> |
| type   | 否    | Array of strings | 保留字段，非必填                                                                                                 |
| desc   | 否    | String           | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为 0--1024<br>最小长度： <b>0</b><br>最大长度： <b>1024</b>                   |
| source | 是    | String           | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度： <b>3</b><br>最大长度： <b>3</b>                                               |

| 参数        | 是否必选 | 参数类型                                | 描述                                                                                                    |
|-----------|------|-------------------------------------|-------------------------------------------------------------------------------------------------------|
| locale    | 是    | String                              | 语言，必填，目前可填zh-cn和en-us<br>枚举值： <ul style="list-style-type: none"><li>• zh-cn</li><li>• en-us</li></ul> |
| templates | 是    | Array of <b>SubTemplate</b> objects | 模板正文，为一个数组                                                                                            |

表 6-515 SubTemplate

| 参数       | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                        |
|----------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | 是    | String | 模板子类型，只支持以下5种类型：<br>sms,dingding,wechat,webhook,email<br>枚举值： <ul style="list-style-type: none"><li>• sms</li><li>• dingding</li><li>• wechat</li><li>• webhook</li><li>• email</li><li>• voice</li></ul> |

| 参数      | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|---------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | 是    | String | <p>子模板正文，\$符号后所跟变量仅支持以下变量，根据不同告警类型（关键词告警和sql告警），所支持的变量亦不相同。目前两种告警类型有共同变量如下：</p> <p>告警级别：\${event_severity};<br/>           发生时间：\${starts_at};<br/>           告警源：<br/>           \$event.metadata.resource_provider;<br/>           资源类型：<br/>           \$event.metadata.resource_type;<br/>           资源标识：\${resources};<br/>           统计类型：关键词统计;<br/>           表达式：<br/>           \$event.annotations.condition_expression;<br/>           当前值：<br/>           \$event.annotations.current_value;<br/>           统计周期：<br/>           \$event.annotations.frequency;<br/>           关键词告警特有变量：<br/>           查询时间：<br/>           \$event.annotations.results[0].time;<br/>           查询日志：<br/>           \$event.annotations.results[0].raw_results;<br/>           sql告警特有变量：<br/>           日志组/流名称：<br/>           \$event.annotations.results[0].resource_id;<br/>           查询语句：<br/>           \$event.annotations.results[0].sql;<br/>           查询时间：<br/>           \$event.annotations.results[0].time;<br/>           查询URL：<br/>           \$event.annotations.results[0].url;查询日志：</p> |

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                                                  |
|-------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------|
|       |      |        | <code>\$event.annotations.results[0].raw_results;</code><br><b>说明</b><br>变量后面的分号";"为英文符号，必须添加，否则模板会出现替换失败的情况<br>最小长度：2<br>最大长度：1024 |
| topic | 否    | String | 邮件主题,只有sub_type=email时生效                                                                                                            |

## 响应参数

状态码：200

表 6-516 响应 Body 参数

| 参数     | 参数类型             | 描述                                                                                                |
|--------|------------------|---------------------------------------------------------------------------------------------------|
| name   | String           | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度：1<br>最大长度：100          |
| type   | Array of strings | 保留字段，非必填                                                                                          |
| desc   | String           | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为0--1024<br>最小长度：0<br>最大长度：1024                             |
| source | String           | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度：3<br>最大长度：3                                                        |
| locale | String           | 语言，必填，目前可填zh-cn和en-us<br>枚举值： <ul style="list-style-type: none"><li>zh-cn</li><li>en-us</li></ul> |

| 参数        | 参数类型                                       | 描述         |
|-----------|--------------------------------------------|------------|
| templates | Array of <b>SubTemplateResBody</b> objects | 模板正文，为一个数组 |

表 6-517 SubTemplateResBody

| 参数       | 参数类型   | 描述                                                                                                                                                                                                                |
|----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | String | 模板子类型，只支持以下6种类型：<br>sms,dingding,wechat,webhook,email<br>枚举值：<br>枚举值： <ul style="list-style-type: none"><li>• sms</li><li>• dingding</li><li>• wechat</li><li>• webhook</li><li>• email</li><li>• voice</li></ul> |

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | String | <p>子模板正文, \$符号后所跟变量仅支持以下变量, 根据不同告警类型(关键词告警和sql告警), 所支持的变量亦不相同。目前两种告警类型有共同变量如下:</p> <p>告警级别: \${event_severity};</p> <p>发生时间: \${starts_at};</p> <p>告警源:<br/>\$event.metadata.resource_provider;</p> <p>资源类型:<br/>\$event.metadata.resource_type;</p> <p>资源标识: \${resources};</p> <p>统计类型: 关键词统计;</p> <p>表达式:<br/>\$event.annotations.condition_expression;</p> <p>当前值:<br/>\$event.annotations.current_value;</p> <p>统计周期:<br/>\$event.annotations.frequency;</p> <p>关键词告警特有变量:</p> <p>查询时间:<br/>\$event.annotations.results[0].time;</p> <p>查询日志:<br/>\$event.annotations.results[0].raw_results;</p> <p>sql告警特有变量:</p> <p>日志组/流名称:<br/>\$event.annotations.results[0].resource_id;</p> <p>查询语句:<br/>\$event.annotations.results[0].sql;</p> <p>查询时间:<br/>\$event.annotations.results[0].time;</p> <p>查询URL:<br/>\$event.annotations.results[0].url;</p> <p>查询日志:<br/>\$event.annotations.results[0].raw_results;</p> <p><b>说明</b><br/>变量后面的分号";"为英文符号, 必须添加, 否则模板会出现替换失败的情况。</p> |
| topic   | String | 邮件主题, 只有sub_type=email时生效                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

**状态码：400****表 6-518 响应 Body 参数**

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**状态码：500****表 6-519 响应 Body 参数**

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**请求示例****创建消息模板**

POST https://{endpoint}/v2/{project\_id}/{domain\_id}/lts/events/notification/templates

```
{
 "name": "xxx",
 "desc": "xxxxxx",
 "type": [],
 "source": "LTS",
 "locale": "zh-cn",
 "templates": [{
 "sub_type": "sms",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "dingding",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "wechat",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "webhook",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;"
 }
]
```

```
}, {
 "sub_type": "email",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n告警源:
${event.metadata.resource_provider};\n资源类型: ${event.metadata.resource_type};\n资源标识: ${resources};\n
统计类型: 关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值:
${event.annotations.current_value};\n统计周期: ${event.annotations.frequency};\n查询时间:
${event.annotations.results[0].time};\n查询日志: ${event.annotations.results[0].raw_results};"
}, {
 "sub_type": "voice",
 "content": "告警级别: ${event_severity}; \n发生时间: ${starts_at}; \n资源标识: ${resources}; \n统计类
型: 关键词统计; \n表达式: ${event.annotations.condition_expression}; \n当前值:
${event.annotations.current_value}; \n统计周期: ${event.annotations.frequency};"
}]
}
```

## 响应示例

### 状态码：200

请求响应成功。

```
{
 "desc": "description",
 "locale": "zh-cn",
 "name": "postman-test",
 "source": "LTS",
 "templates": [{
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "sms"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "dingding"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "wechat"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "webhook"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n告警源:
${event.metadata.resource_provider};\n资源类型: ${event.metadata.resource_type};\n资源标识: ${resources};\n
统计类型: 关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值:
${event.annotations.current_value};\n统计周期: ${event.annotations.frequency};\n查询时间:
${event.annotations.results[0].time};\n查询日志: ${event.annotations.results[0].raw_results};",
 "sub_type": "email"
 }, {
 "content": "告警级别: ${event_severity}; \n发生时间: ${starts_at}; \n资源标识: ${resources}; \n统计类
型: 关键词统计; \n表达式: ${event.annotations.condition_expression}; \n当前值:
${event.annotations.current_value}; \n统计周期: ${event.annotations.frequency};",
 "sub_type": "voice"
 }],
 "type": []
}
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.2014",
```

```
"error_msg" : "desc is invalid!"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2014",
 "error_msg" : "Failed to create notification template."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 创建消息模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateNotificationTemplateSolution {
 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateNotificationTemplateRequest request = new CreateNotificationTemplateRequest();
 request.withDomainId("{domain_id}");
 CreateNotificationTemplateRequestBody body = new CreateNotificationTemplateRequestBody();
 List<SubTemplate> listbodyTemplates = new ArrayList<>();
 listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("sms"))
 .withContent("告警级别:{event_severity};
发生时间:{starts_at};
资源标识:{resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;")
```

```
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("dingding"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("wechat"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("webhook"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("email"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
告警源:${event.metadata.resource_provider};
资源类型:${event.metadata.resource_type};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};
查询时间:${event.annotations.results[0].time};
查询日志:${event.annotations.results[0].raw_results};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("voice"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
body.withTemplates(listbodyTemplates);
body.withLocale(CreateNotificationTemplateRequestBody.LocaleEnum.fromValue("zh-cn"));
body.withSource("LTS");
body.withDesc("xxxxxx");
body.withName("xxx");
request.withBody(body);
try {
 CreateNotificationTemplateResponse response = client.createNotificationTemplate(request);
 System.out.println(response.toString());
}
```

```
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 创建消息模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateNotificationTemplateRequest()
 request.domain_id = "{domain_id}"
 listTemplatesbody = [
 SubTemplate(
 sub_type="sms",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="dingding",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="wechat",
 content="告警级别:${event_severity};"
```

```
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="webhook",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="email",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 告警源:${event.metadata.resource_provider};
 资源类型:${event.metadata.resource_type};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:${event.annotations.frequency};
 查询时间:${event.annotations.results[0].time};
 查询日志:${event.annotations.results[0].raw_results;"
),
 SubTemplate(
 sub_type="voice",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:${event.annotations.frequency};"
)
]
request.body = CreateNotificationTemplateRequestBody(
 templates=listTemplatesbody,
 locale="zh-cn",
 source="LTS",
 desc="xxxxxx",
 name="xxx"
)
response = client.create_notification_template(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建消息模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
```

```
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateNotificationTemplateRequest{
 request.DomainId = "{domain_id}"
 }
 var listTemplatesbody = []model.SubTemplate{
 {
 SubType: model.GetSubTemplateSubTypeEnum().SMS,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().DINGDING,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().WECHAT,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().WEBHOOK,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().EMAIL,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
```

```
告警源:$event.metadata.resource_provider;
资源类型:$event.metadata.resource_type;
资源标识:${resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;
查询时间:$event.annotations.results[0].time;
查询日志:$event.annotations.results[0].raw_results;",
},
{
 SubType: model.GetSubTemplateSubTypeEnum().VOICE,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;",
},
}
descCreateNotificationTemplateRequestBody:= "xxxxxx"
request.Body = &model.CreateNotificationTemplateRequestBody{
 Templates: listTemplatesbody,
 Locale: model.GetCreateNotificationTemplateRequestBodyLocaleEnum().ZH_CN,
 Source: "LTS",
 Desc: &descCreateNotificationTemplateRequestBody,
 Name: "xxx",
}
response, err := client.CreateNotificationTemplate(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

### 6.11.2 修改消息模板

#### 功能介绍

该接口用于修改通知模板,根据名称进行修改。

## 调用方法

请参见[如何调用API](#)。

## URI

PUT /v2/{project\_id}/{domain\_id}/lts/events/notification/templates

表 6-520 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id  | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-521 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-522 请求 Body 参数

| 参数        | 是否必选 | 参数类型                               | 描述                                                                                                          |
|-----------|------|------------------------------------|-------------------------------------------------------------------------------------------------------------|
| name      | 是    | String                             | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度：1<br>最大长度：100                    |
| type      | 否    | Array of strings                   | 保留字段，非必填                                                                                                    |
| desc      | 否    | String                             | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为 0--1024<br>最小长度：0<br>最大长度：1024                                      |
| source    | 是    | String                             | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度：3<br>最大长度：3                                                                  |
| locale    | 是    | String                             | 语言，必填，目前可填zh-cn和en-us<br>枚举值：<br><ul style="list-style-type: none"> <li>• zh-cn</li> <li>• en-us</li> </ul> |
| templates | 是    | Array of UpdateSubTemplate objects | 模板正文，为一个数组                                                                                                  |

表 6-523 UpdateSubTemplate

| 参数       | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                              |
|----------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | 是    | String | 模板子类型，只支持以下6种类型：<br>sms,dingding,wechat,webhook,email,voice<br>枚举值： <ul style="list-style-type: none"><li>• sms</li><li>• dingding</li><li>• wechat</li><li>• webhook</li><li>• email</li><li>• voice</li></ul> |

| 参数      | 是否必选 | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | 是    | String | <p>子模板正文，\$符号后所跟变量仅支持以下变量，根据不同告警类型（关键词告警和sql告警），所支持的变量亦不相同。目前两种告警类型有共同变量如下：</p> <p>告警级别：\${event_severity};</p> <p>发生时间：\${starts_at};</p> <p>告警源：<br/>\$event.metadata.resource_provider;</p> <p>资源类型：<br/>\$event.metadata.resource_type;</p> <p>资源标识：\${resources};</p> <p>统计类型：关键词统计;</p> <p>表达式：<br/>\$event.annotations.condition_expression;</p> <p>当前值：<br/>\$event.annotations.current_value;</p> <p>统计周期：<br/>\$event.annotations.frequency;</p> <p>关键词告警特有变量：</p> <p>查询时间：<br/>\$event.annotations.results[0].time;</p> <p>查询日志：<br/>\$event.annotations.results[0].raw_results;</p> <p>sql告警特有变量：</p> <p>日志组/流名称：<br/>\$event.annotations.results[0].resource_id;</p> <p>查询语句：<br/>\$event.annotations.results[0].sql;</p> <p>查询时间：<br/>\$event.annotations.results[0].time;</p> <p>查询URL：<br/>\$event.annotations.results[0].url;</p> |

| 参数    | 是否必选 | 参数类型   | 描述                                                                                                        |
|-------|------|--------|-----------------------------------------------------------------------------------------------------------|
|       |      |        | 查询日志：<br>\$event.annotations.results[0].raw_results;<br><b>说明</b><br>变量后面的分号";"为英文符号，必须添加，否则模板会出现替换失败的情况。 |
| topic | 否    | String | 邮件主题,只有sub_type=email时生效                                                                                  |

## 响应参数

状态码：201

表 6-524 响应 Body 参数

| 参数        | 参数类型                                                 | 描述                                                                                                       |
|-----------|------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| name      | String                                               | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度：1<br>最大长度：100                 |
| type      | Array of strings                                     | 保留字段，非必填                                                                                                 |
| desc      | String                                               | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为0--1024<br>最小长度：0<br>最大长度：1024                                    |
| source    | String                                               | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度：3<br>最大长度：3                                                               |
| locale    | String                                               | 语言，必填，目前可填zh-cn和en-us<br>枚举值： <ul style="list-style-type: none"> <li>• zh-cn</li> <li>• en-us</li> </ul> |
| templates | Array of <a href="#">SubTemplateRes Body</a> objects | 模板正文，为一个数组                                                                                               |

表 6-525 SubTemplateResBody

| 参数       | 参数类型   | 描述                                                                                                                                                                                                                   |
|----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | String | 模板子类型，只支持以下6种类型：<br>sms,dingding,wechat,webhook,email<br>枚举值：<br>枚举值：<br><ul style="list-style-type: none"><li>• sms</li><li>• dingding</li><li>• wechat</li><li>• webhook</li><li>• email</li><li>• voice</li></ul> |

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | String | <p>子模板正文, \$符号后所跟变量仅支持以下变量, 根据不同告警类型(关键词告警和sql告警), 所支持的变量亦不相同。目前两种告警类型有共同变量如下:</p> <p>告警级别: <code>\${event_severity}</code>;</p> <p>发生时间: <code>\${starts_at}</code>;</p> <p>告警源:<br/><code>\$event.metadata.resource_provider</code>;</p> <p>资源类型:<br/><code>\$event.metadata.resource_type</code>;</p> <p>资源标识: <code>\${resources}</code>;</p> <p>统计类型: 关键词统计;</p> <p>表达式:<br/><code>\$event.annotations.condition_expression</code>;</p> <p>当前值:<br/><code>\$event.annotations.current_value</code>;</p> <p>统计周期:<br/><code>\$event.annotations.frequency</code>;</p> <p>关键词告警特有变量:</p> <p>查询时间:<br/><code>\$event.annotations.results[0].time</code>;</p> <p>查询日志:<br/><code>\$event.annotations.results[0].raw_results</code>;</p> <p>sql告警特有变量:</p> <p>日志组/流名称:<br/><code>\$event.annotations.results[0].resource_id</code>;</p> <p>查询语句:<br/><code>\$event.annotations.results[0].sql</code>;</p> <p>查询时间:<br/><code>\$event.annotations.results[0].time</code>;</p> <p>查询URL:<br/><code>\$event.annotations.results[0].url</code>;</p> <p>查询日志:<br/><code>\$event.annotations.results[0].raw_results</code>;</p> <p><b>说明</b><br/>变量后面的分号";"为英文符号, 必须添加, 否则模板会出现替换失败的情况。</p> |
| topic   | String | 邮件主题, 只有sub_type=email时生效                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

**状态码：400**

**表 6-526 响应 Body 参数**

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**状态码：500**

**表 6-527 响应 Body 参数**

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**请求示例**

修改消息模板

PUT https://{endpoint}/v2/{project\_id}/{domain\_id}/lts/events/notification/templates

```
{
 "name": "xxx",
 "desc": "xxxxxx",
 "type": [],
 "source": "LTS",
 "locale": "zh-cn",
 "templates": [{
 "sub_type": "sms",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型: 关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "dingding",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型: 关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "wechat",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型: 关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n统计周期: $event.annotations.frequency;"
 }, {
 "sub_type": "webhook",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型: 关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n统计周期: $event.annotations.frequency;"
 }
]
```

```
}, {
 "sub_type": "email",
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n告警源:
${event.metadata.resource_provider};\n资源类型: ${event.metadata.resource_type};\n资源标识: ${resources};\n
统计类型: 关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值:
${event.annotations.current_value};\n统计周期: ${event.annotations.frequency};\n查询时间:
${event.annotations.results[0].time};\n查询日志: ${event.annotations.results[0].raw_results};"
}, {
 "sub_type": "voice",
 "content": "告警级别: ${event_severity}; \n发生时间: ${starts_at}; \n资源标识: ${resources}; \n统计类
型: 关键词统计; \n表达式: ${event.annotations.condition_expression}; \n当前值:
${event.annotations.current_value}; \n统计周期: ${event.annotations.frequency};"
}]
}
```

## 响应示例

### 状态码：201

请求响应成功。

```
{
 "desc": "description",
 "locale": "zh-cn",
 "name": "postman-test1",
 "source": "LTS",
 "templates": [{
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "sms"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "dingding"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "wechat"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值: ${event.annotations.current_value};\n
统计周期: ${event.annotations.frequency};",
 "sub_type": "webhook"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n告警源:
${event.metadata.resource_provider};\n资源类型: ${event.metadata.resource_type};\n资源标识: ${resources};\n
统计类型: 关键词统计;\n表达式: ${event.annotations.condition_expression};\n当前值:
${event.annotations.current_value};\n统计周期: ${event.annotations.frequency};\n查询时间:
${event.annotations.results[0].time};\n查询日志: ${event.annotations.results[0].raw_results};",
 "sub_type": "email"
 }, {
 "content": "告警级别: ${event_severity}; \n发生时间: ${starts_at}; \n资源标识: ${resources}; \n统计类
型: 关键词统计; \n表达式: ${event.annotations.condition_expression}; \n当前值:
${event.annotations.current_value}; \n统计周期: ${event.annotations.frequency};",
 "sub_type": "voice"
 }],
 "type": []
}
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.2016",
```

```
"error_msg" : "desc is invalid!"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2016",
 "error_msg" : "Failed to update notification template"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 修改消息模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateNotificationTemplateSolution {
 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateNotificationTemplateRequest request = new UpdateNotificationTemplateRequest();
 request.withDomainId("{domain_id}");
 CreateNotificationTemplateRequestBody body = new CreateNotificationTemplateRequestBody();
 List<SubTemplate> listbodyTemplates = new ArrayList<>();
 listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("sms"))
 .withContent("告警级别:{event_severity};
发生时间:{starts_at};
资源标识:{resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;")
```

```
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("dingding"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("wechat"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("webhook"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("email"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
告警源:${event.metadata.resource_provider};
资源类型:${event.metadata.resource_type};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};
查询时间:${event.annotations.results[0].time};
查询日志:${event.annotations.results[0].raw_results};")
);
listbodyTemplates.add(
 new SubTemplate()
 .withSubType(SubTemplate.SubTypeEnum.fromValue("voice"))
 .withContent("告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};")
);
body.withTemplates(listbodyTemplates);
body.withLocale(CreateNotificationTemplateRequestBody.LocaleEnum.fromValue("zh-cn"));
body.withSource("LTS");
body.withDesc("xxxxxx");
body.withName("xxx");
request.withBody(body);
try {
 UpdateNotificationTemplateResponse response = client.updateNotificationTemplate(request);
 System.out.println(response.toString());
}
```

```
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 修改消息模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateNotificationTemplateRequest()
 request.domain_id = "{domain_id}"
 listTemplatesbody = [
 SubTemplate(
 sub_type="sms",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="dingding",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};"
),
 SubTemplate(
 sub_type="wechat",
 content="告警级别:${event_severity};"
```

```
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:$event.annotations.frequency;"
),
 SubTemplate(
 sub_type="webhook",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:$event.annotations.frequency;"
),
 SubTemplate(
 sub_type="email",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 告警源:$event.metadata.resource_provider;
 资源类型:$event.metadata.resource_type;
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:$event.annotations.frequency;
 查询时间:$event.annotations.results[0].time;
 查询日志:$event.annotations.results[0].raw_results;"
),
 SubTemplate(
 sub_type="voice",
 content="告警级别:${event_severity};
 发生时间:${starts_at};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: $event.annotations.current_value;
 统计周期:$event.annotations.frequency;"
)
]
request.body = CreateNotificationTemplateRequestBody(
 templates=listTemplatesbody,
 locale="zh-cn",
 source="LTS",
 desc="xxxxxx",
 name="xxx"
)
response = client.update_notification_template(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 修改消息模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
```

```
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateNotificationTemplateRequest{}
 request.DomainId = "{domain_id}"
 var listTemplatesbody = []model.SubTemplate{
 {
 SubType: model.GetSubTemplateSubTypeEnum().SMS,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().DINGDING,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().WECHAT,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().WEBHOOK,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:${event.annotations.condition_expression};
当前值: ${event.annotations.current_value};
统计周期:${event.annotations.frequency};",
 },
 {
 SubType: model.GetSubTemplateSubTypeEnum().EMAIL,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
```

```
告警源:$event.metadata.resource_provider;
资源类型:$event.metadata.resource_type;
资源标识:${resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;
查询时间:$event.annotations.results[0].time;
查询日志:$event.annotations.results[0].raw_results;";
},
{
 SubType: model.GetSubTemplateSubTypeEnum().VOICE,
 Content: "告警级别:${event_severity};
发生时间:${starts_at};
资源标识:${resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;";
},
}
}
descCreateNotificationTemplateRequestBody:= "xxxxxx"
request.Body = &model.CreateNotificationTemplateRequestBody{
 Templates: listTemplatesbody,
 Locale: model.GetCreateNotificationTemplateRequestBodyLocaleEnum().ZH_CN,
 Source: "LTS",
 Desc: &descCreateNotificationTemplateRequestBody,
 Name: "xxx",
}
response, err := client.UpdateNotificationTemplate(request)
if err == nil {
 fmt.Printf("%v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 201 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

### 6.11.3 查询消息模板列表

#### 功能介绍

该接口用于查询通知模板列表。

## 调用方法

请参见[如何调用API](#)。

## URI

GET /v2/{project\_id}/{domain\_id}/lts/events/notification/templates

表 6-528 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id  | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-529 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-530 响应 Body 参数

| 参数      | 参数类型                                         | 描述   |
|---------|----------------------------------------------|------|
| results | Array of <b>NotificationTemplate</b> objects | 模板数组 |

表 6-531 NotificationTemplate

| 参数          | 参数类型                                            | 描述                                                                                                    |
|-------------|-------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| name        | String                                          | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度：1<br>最大长度：100              |
| type        | Array of strings                                | 保留字段，非必填                                                                                              |
| desc        | String                                          | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为0--1024<br>最小长度：0<br>最大长度：1024                                 |
| source      | String                                          | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度：3<br>最大长度：3                                                            |
| locale      | String                                          | 语言，必填，目前可填zh-cn和en-us<br>枚举值： <ul style="list-style-type: none"><li>• zh-cn</li><li>• en-us</li></ul> |
| templates   | Array of <b>SubTemplateResponseBody</b> objects | 模板正文，为一个数组                                                                                            |
| create_time | Long                                            | 创建时间，为毫秒时间戳<br>最小值：0<br>最大值：1700000000000                                                             |
| modify_time | Long                                            | 更新时间，为毫秒时间戳<br>最小值：0<br>最大值：1700000000000                                                             |

| 参数         | 参数类型   | 描述                                                                            |
|------------|--------|-------------------------------------------------------------------------------|
| project_id | String | 项目ID，获取方式请参见：获取账号ID、项目ID、日志组ID、日志流ID（lts_api_0006.xml）。<br>最小长度：32<br>最大长度：32 |

表 6-532 SubTemplateResBody

| 参数       | 参数类型   | 描述                                                                                                                                                                                                                   |
|----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | String | 模板子类型，只支持以下6种类型：<br>sms,dingding,wechat,webhook,email<br>枚举值：<br>枚举值：<br><ul style="list-style-type: none"><li>• sms</li><li>• dingding</li><li>• wechat</li><li>• webhook</li><li>• email</li><li>• voice</li></ul> |

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | String | <p>子模板正文, \$符号后所跟变量仅支持以下变量, 根据不同告警类型(关键词告警和sql告警), 所支持的变量亦不相同。目前两种告警类型有共同变量如下:</p> <p>告警级别: <code>\${event_severity}</code>;</p> <p>发生时间: <code>\${starts_at}</code>;</p> <p>告警源:<br/><code>\$event.metadata.resource_provider</code>;</p> <p>资源类型:<br/><code>\$event.metadata.resource_type</code>;</p> <p>资源标识: <code>\${resources}</code>;</p> <p>统计类型: 关键词统计;</p> <p>表达式:<br/><code>\$event.annotations.condition_expression</code>;</p> <p>当前值:<br/><code>\$event.annotations.current_value</code>;</p> <p>统计周期:<br/><code>\$event.annotations.frequency</code>;</p> <p>关键词告警特有变量:</p> <p>查询时间:<br/><code>\$event.annotations.results[0].time</code>;</p> <p>查询日志:<br/><code>\$event.annotations.results[0].raw_results</code>;</p> <p>sql告警特有变量:</p> <p>日志组/流名称:<br/><code>\$event.annotations.results[0].resource_id</code>;</p> <p>查询语句:<br/><code>\$event.annotations.results[0].sql</code>;</p> <p>查询时间:<br/><code>\$event.annotations.results[0].time</code>;</p> <p>查询URL:<br/><code>\$event.annotations.results[0].url</code>;</p> <p>查询日志:<br/><code>\$event.annotations.results[0].raw_results</code>;</p> <p><b>说明</b><br/>变量后面的分号";"为英文符号, 必须添加, 否则模板会出现替换失败的情况。</p> |
| topic   | String | 邮件主题, 只有sub_type=email时生效                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

**状态码：500**

表 6-533 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**请求示例**

## 查询消息模板列表

```
GET https://{endpoint}/v2/{project_id}/{domain_id}/lts/events/notification/templates
/v2/{project_id}/{domain_id}/lts/events/notification/templates
```

**响应示例****状态码：200**

请求响应成功。

```
{
 "results": [{
 "create_time": 1701352010150,
 "desc": "这是短信测试模式",
 "locale": "zh-cn",
 "modify_time": 1701352010150,
 "name": "15nWzUsOHA",
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "source": "LTS",
 "templates": [{
 "content": "这是一个短信测试模板",
 "sub_type": "sms"
 }],
 "type": [""]
 }, {
 "create_time": 1702021411612,
 "desc": "这是短信测试模式",
 "locale": "zh-cn",
 "modify_time": 1702021411612,
 "name": "RZ2ObeluNN",
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "source": "LTS",
 "templates": [{
 "content": "这是一个短信测试模板",
 "sub_type": "sms"
 }],
 "type": [""]
 }]
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2017",
```

```
"error_msg": "Find Alarm rule failed."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.11.4 删除消息模板

### 功能介绍

该接口用于删除通知模板。

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v2/{project\_id}/{domain\_id}/lts/events/notification/templates

表 6-534 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id  | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-535 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-536 请求 Body 参数

| 参数             | 是否必选 | 参数类型             | 描述        |
|----------------|------|------------------|-----------|
| template_names | 是    | Array of strings | 待删除模板名称数组 |

## 响应参数

状态码：400

表 6-537 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

状态码：500

表 6-538 响应 Body 参数

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

## 请求示例

删除消息模板

```
DELETE https://{endpoint}/v2/{project_id}/{domain_id}/lts/events/notification/templates
/v2/{project_id}/{domain_id}/lts/events/notification/templates
{"template_names":["template1","template2"]}
```

## 响应示例

**状态码：400**

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.2015",
 "error_msg" : "delete template name is empty or projectId is null"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2015",
 "error_msg" : "Failed to delete notification template."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

删除消息模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class DeleteNotificationTemplateSolution {
```

```
public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 DeleteNotificationTemplateRequest request = new DeleteNotificationTemplateRequest();
 request.withDomainId("{domain_id}");
 try {
 DeleteNotificationTemplateResponse response = client.deleteNotificationTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrMsg());
 }
}
```

## Python

### 删除消息模板

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = DeleteNotificationTemplateRequest()
 request.domain_id = "{domain_id}"
 response = client.delete_notification_template(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 删除消息模板

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.DeleteNotificationTemplateRequest{}
 request.DomainId = "{domain_id}"
 response, err := client.DeleteNotificationTemplate(request)
 if err == nil {
 fmt.Printf("%v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 204 | 响应体为空，请求响应成功。                         |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |

| 状态码 | 描述                     |
|-----|------------------------|
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.11.5 查询单个消息模板

### 功能介绍

该接口用于查询单个通知模板

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/{domain\_id}/lts/events/notification/template/  
{template\_name}

表 6-539 路径参数

| 参数            | 是否必选 | 参数类型   | 描述                                                                            |
|---------------|------|--------|-------------------------------------------------------------------------------|
| project_id    | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id     | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| template_name | 是    | String | 需要获取的消息模板名称<br>最小长度：1<br>最大长度：100                                             |

## 请求参数

表 6-540 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-541 响应 Body 参数

| 参数     | 参数类型             | 描述                                                                                       |
|--------|------------------|------------------------------------------------------------------------------------------|
| name   | String           | 通知规则名称，必填，只含有汉字、数字、字母、下划线、中划线，不能以下划线等特殊符号开头和结尾，长度为 1 - 100，创建后不可修改<br>最小长度：1<br>最大长度：100 |
| type   | Array of strings | 保留字段，非必填                                                                                 |
| desc   | String           | 模板描述，非必填，只含有汉字、数字、字母、下划线不能以下划线开头和结尾，长度为0--1024<br>最小长度：0<br>最大长度：1024                    |
| source | String           | 模板来源，目前必填为LTS，否则会筛选不出来<br>最小长度：3<br>最大长度：3                                               |

| 参数          | 参数类型                                                | 描述                                                                                                      |
|-------------|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| locale      | String                                              | 语言，必填，目前可填zh-cn和en-us<br>枚举值：<br><ul style="list-style-type: none"> <li>zh-cn</li> <li>en-us</li> </ul> |
| templates   | Array of <a href="#">SubTemplateResBody</a> objects | 模板正文，为一个数组                                                                                              |
| create_time | Long                                                | 创建时间，为毫秒时间戳<br>最小值：0<br>最大值：1700000000000                                                               |
| modify_time | Long                                                | 更新时间，为毫秒时间戳<br>最小值：0<br>最大值：1700000000000                                                               |
| project_id  | String                                              | 项目ID，获取方式请参见：获取账号ID、项目ID、日志组ID、日志流ID（lts_api_0006.xml）。<br>最小长度：32<br>最大长度：32                           |

表 6-542 SubTemplateResBody

| 参数       | 参数类型   | 描述                                                                                                                                                                                                              |
|----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_type | String | 模板子类型，只支持以下6种类型：<br>sms,dingding,wechat,webhook,email<br>枚举值：<br>枚举值：<br><ul style="list-style-type: none"> <li>sms</li> <li>dingding</li> <li>wechat</li> <li>webhook</li> <li>email</li> <li>voice</li> </ul> |

| 参数      | 参数类型   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content | String | <p>子模板正文, \$符号后所跟变量仅支持以下变量, 根据不同告警类型(关键词告警和sql告警), 所支持的变量亦不相同。目前两种告警类型有共同变量如下:</p> <p>告警级别: \${event_severity};</p> <p>发生时间: \${starts_at};</p> <p>告警源:<br/>\$event.metadata.resource_provider;</p> <p>资源类型:<br/>\$event.metadata.resource_type;</p> <p>资源标识: \${resources};</p> <p>统计类型: 关键词统计;</p> <p>表达式:<br/>\$event.annotations.condition_expression;</p> <p>当前值:<br/>\$event.annotations.current_value;</p> <p>统计周期:<br/>\$event.annotations.frequency;</p> <p>关键词告警特有变量:</p> <p>查询时间:<br/>\$event.annotations.results[0].time;</p> <p>查询日志:<br/>\$event.annotations.results[0].raw_results;</p> <p>sql告警特有变量:</p> <p>日志组/流名称:<br/>\$event.annotations.results[0].resource_id;</p> <p>查询语句:<br/>\$event.annotations.results[0].sql;</p> <p>查询时间:<br/>\$event.annotations.results[0].time;</p> <p>查询URL:<br/>\$event.annotations.results[0].url;</p> <p>查询日志:<br/>\$event.annotations.results[0].raw_results;</p> <p><b>说明</b><br/>变量后面的分号";"为英文符号, 必须添加, 否则模板会出现替换失败的情况。</p> |
| topic   | String | 邮件主题, 只有sub_type=email时生效                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

**状态码：500****表 6-543 响应 Body 参数**

| 参数         | 参数类型   | 描述                       |
|------------|--------|--------------------------|
| error_code | String | 错误码。<br>最小长度：8<br>最大长度：8 |
| error_msg  | String | 调用失败响应信息描述。              |

**请求示例**

查询指定名称的消息模板

```
GET https://{endpoint}/v2/{project_id}/{domain_id}/lts/events/notification/template/{template_name}
/v2/{project_id}/{domain_id}/lts/events/notification/template/{template_name}
```

**响应示例****状态码：200**

请求响应成功。

```
{
 "create_time": 1702955600631,
 "desc": "description",
 "locale": "zh-cn",
 "modify_time": 1702955600631,
 "name": "postman-test",
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "source": "LTS",
 "templates": [{
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;";
 "sub_type": "sms"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;";
 "sub_type": "dingding"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;";
 "sub_type": "wechat"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n资源标识: ${resources};\n统计类型:
关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值: $event.annotations.current_value;\n
统计周期: $event.annotations.frequency;";
 "sub_type": "webhook"
 }, {
 "content": "告警级别: ${event_severity};\n发生时间: ${starts_at};\n告警源:
$event.metadata.resource_provider;\n资源类型: $event.metadata.resource_type;\n资源标识: ${resources};\n
统计类型: 关键词统计;\n表达式: $event.annotations.condition_expression;\n当前值:
$event.annotations.current_value;\n统计周期: $event.annotations.frequency;\n查询时间:
$event.annotations.results[0].time;\n查询日志: $event.annotations.results[0].raw_results;";
 "sub_type": "email"
 }, {
 }, {
}
```

```
"content": "告警级别: ${event_severity}; \n发生时间: ${starts_at}; \n资源标识: ${resources}; \n统计类型: 关键词统计; \n表达式: $event.annotations.condition_expression; \n当前值: $event.annotations.current_value; \n统计周期: $event.annotations.frequency;",
 "sub_type": "voice"
}],
"type": []
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2018",
 "error_msg": "Failed to get notification template."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.11.6 预览消息模板邮件格式

### 功能介绍

该接口用于预览通知模板邮件格式

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/{domain\_id}/lts/events/notification/templates/view

表 6-544 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

| 参数        | 是否必选 | 参数类型   | 描述                                                                            |
|-----------|------|--------|-------------------------------------------------------------------------------|
| domain_id | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-545 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-546 请求 Body 参数

| 参数        | 是否必选 | 参数类型   | 描述                                                                                                       |
|-----------|------|--------|----------------------------------------------------------------------------------------------------------|
| templates | 是    | String | 邮件模板内容<br>最小长度：2<br>最大长度：1024                                                                            |
| language  | 是    | String | 语言 zh-cn中文，en-us英文<br>枚举值：<br><ul style="list-style-type: none"> <li>• zh-cn</li> <li>• en-us</li> </ul> |
| source    | 是    | String | 来源，只能填LTS<br>最小长度：3<br>最大长度：3                                                                            |

## 响应参数

状态码：200



```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListNotificationTemplateSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListNotificationTemplateRequest request = new ListNotificationTemplateRequest();
 request.withDomainId("{domain_id}");
 PreviewTemplateBody body = new PreviewTemplateBody();
 body.withSource("LTS");
 body.withLanguage(PreviewTemplateBody.LanguageEnum.fromValue("zh-cn"));
 body.withTemplates("告警级别:${event_severity};
 发生时间:${starts_at};
 告警源:${event.metadata.resource_provider};
 资源类型:${event.metadata.resource_type};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};
 查询时间:${event.annotations.results[0].time};
 查询日志:${event.annotations.results[0].raw_results;}");
 request.withBody(body);
 try {
 ListNotificationTemplateResponse response = client.listNotificationTemplate(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

### 预览消息模板邮件格式

```
coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListNotificationTemplateRequest()
 request.domain_id = "{domain_id}"
 request.body = PreviewTemplateBody(
 source="LTS",
 language="zh-cn",
 templates="告警级别:${event_severity};
 发生时间:${starts_at};
 告警源:${event.metadata.resource_provider};
 资源类型:${event.metadata.resource_type};
 资源标识:${resources};
 统计类型:关键词统计;
 表达式:${event.annotations.condition_expression};
 当前值: ${event.annotations.current_value};
 统计周期:${event.annotations.frequency};
 查询时间:${event.annotations.results[0].time};
 查询日志:${event.annotations.results[0].raw_results;"
)
 response = client.list_notification_template(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 预览消息模板邮件格式

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
```

```
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.ListNotificationTemplateRequest{}
request.DomainId = "{domain_id}"
request.Body = &model.PreviewTemplateBody{
 Source: "LTS",
 Language: model.GetPreviewTemplateBodyLanguageEnum().ZH_CN,
 Templates: "告警级别:${event_severity};
发生时间:${starts_at};
告警源:$event.metadata.resource_provider;
资源类型:$event.metadata.resource_type;
资源标识:${resources};
统计类型:关键词统计;
表达式:$event.annotations.condition_expression;
当前值: $event.annotations.current_value;
统计周期:$event.annotations.frequency;
查询时间:$event.annotations.results[0].time;
查询日志:$event.annotations.results[0].raw_results;"
}
response, err := client.ListNotificationTemplate(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述      |
|-----|---------|
| 200 | 请求响应成功。 |

## 错误码

请参见[错误码](#)。

## 6.12 SQL 告警规则

## 6.12.1 创建 SQL 告警规则

### 功能介绍

该接口用于创建SQL告警，目前每个账户最多可以创建共200个关键词告警与SQL告警

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/lts/alarms/sql-alarm-rule

表 6-548 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-549 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-550 请求 Body 参数

| 参数                          | 是否必选 | 参数类型                                               | 描述                                                                                                                             |
|-----------------------------|------|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| sql_alarm_rule_name         | 是    | String                                             | SQL告警名称<br>最小长度：1<br>最大长度：64                                                                                                   |
| is_css_sql                  | 否    | Boolean                                            | 是否管道符sql查询。<br>缺省值： <b>false</b>                                                                                               |
| sql_alarm_rule_description  | 否    | String                                             | SQL告警信息描述<br>最小长度：0<br>最大长度：64                                                                                                 |
| sql_requests                | 是    | Array of <a href="#">SqlRequest</a> objects        | SQL详细信息                                                                                                                        |
| frequency                   | 是    | <a href="#">CreateSqlAlarmRuleFrequency</a> object | 告警统计周期                                                                                                                         |
| condition_expression        | 是    | String                                             | 条件表达式<br>最小长度：1<br>最大长度：1024                                                                                                   |
| sql_alarm_level             | 是    | String                                             | 告警级别<br>枚举值：<br><ul style="list-style-type: none"> <li>• Info</li> <li>• Minor</li> <li>• Major</li> <li>• Critical</li> </ul> |
| sql_alarm_send              | 是    | Boolean                                            | 是否发送                                                                                                                           |
| domain_id                   | 是    | String                                             | domainId<br>最小长度：32<br>最大长度：32                                                                                                 |
| notification_save_rule      | 否    | <a href="#">SqlNotificationSaveRule</a> object     | 通知主题                                                                                                                           |
| trigger_condition_count     | 否    | Integer                                            | 触发条件：触发次数；默认为1                                                                                                                 |
| trigger_condition_frequency | 否    | Integer                                            | 触发条件：触发周期；默认为1                                                                                                                 |

| 参数                      | 是否必选 | 参数类型    | 描述                                                                                                                                                                           |
|-------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| whether_recovery_policy | 否    | Boolean | 是否打开恢复通知；默认false                                                                                                                                                             |
| recovery_policy         | 否    | Integer | 恢复策略周期；默认为3                                                                                                                                                                  |
| notification_frequency  | 是    | Integer | 通知频率,单位(分钟)<br>枚举值: <ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name  | 否    | String  | 告警行动规则名称<br><b>说明</b><br>alarm_action_rule_name和notification_save_rule可以选填一个，如果都填，优先选择alarm_action_rule_name                                                                 |

表 6-551 SqlRequest

| 参数                     | 是否必选 | 参数类型    | 描述                                          |
|------------------------|------|---------|---------------------------------------------|
| is_time_range_relative | 否    | Boolean | 是时间范围相对                                     |
| log_stream_id          | 是    | String  | 日志流id<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| log_stream_name        | 否    | String  | 日志流名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>  |
| log_group_id           | 是    | String  | 日志组id<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                     |
|------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------|
| log_group_name         | 否    | String  | 日志组名称<br>最小长度：1<br>最大长度：64                                                                                             |
| sql                    | 是    | String  | sql语句<br>最小长度：1<br>最大长度：1024                                                                                           |
| sql_request_title      | 是    | String  | 图表名称<br>最小长度：1<br>最大长度：64                                                                                              |
| search_time_range      | 是    | Integer | 查询执行任务时最近数据的时间范围（当 search_time_range_unit 为 minute，则最大值为60；当 search_time_range_unit 为 hour，则最大值为24）<br>最小值：1<br>最大值：60 |
| search_time_range_unit | 是    | String  | 查询时间单位<br>枚举值：<br><ul style="list-style-type: none"> <li>• minute</li> <li>• hour</li> </ul>                           |

表 6-552 CreateSqlAlarmRuleFrequency

| 参数        | 是否必选 | 参数类型   | 描述                                                                                                                                                   |
|-----------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| type      | 是    | String | 时间类型。<br>枚举值：<br><ul style="list-style-type: none"> <li>• CRON</li> <li>• HOURLY</li> <li>• DAILY</li> <li>• WEEKLY</li> <li>• FIXED_RATE</li> </ul> |
| cron_expr | 否    | String | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                          |

| 参数              | 是否必选 | 参数类型    | 描述                                                                                                                |
|-----------------|------|---------|-------------------------------------------------------------------------------------------------------------------|
| hour_of_day     | 否    | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。<br>DAILY: 最小值: 0, 最大值: 23<br>WEEKLY: 最小值: 0, 最大值: 23                              |
| day_of_week     | 否    | Integer | 当字段type为"WEEKLY"时取该字段(周日~周六)。                                                                                     |
| fixed_rate      | 否    | Integer | 当字段type为"FIXED_RATE"时取该字段(当fixed_rate_unit单位为minute, 最大值60; 当fixed_rate_unit单位为hour, 最大值24)。<br>最小值: 1<br>最大值: 60 |
| fixed_rate_unit | 否    | String  | 时间单位。<br>枚举值:<br><ul style="list-style-type: none"> <li>• minute</li> <li>• hour</li> </ul>                       |

表 6-553 SqlNotificationSaveRule

| 参数        | 是否必选 | 参数类型                                    | 描述                                                                                                                    |
|-----------|------|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| language  | 是    | String                                  | 首选项对应的语言<br>最小长度: 0<br>最大长度: 10<br>枚举值:<br><ul style="list-style-type: none"> <li>• zh-cn</li> <li>• en-us</li> </ul> |
| timezone  | 否    | String                                  | 首选项对应的时区信息<br>最小长度: 0<br>最大长度: 1024                                                                                   |
| user_name | 是    | String                                  | 用户名<br>最小长度: 1<br>最大长度: 1024                                                                                          |
| topics    | 是    | Array of <a href="#">Topics</a> objects | 主题信息                                                                                                                  |

| 参数            | 是否必选 | 参数类型   | 描述     |
|---------------|------|--------|--------|
| template_name | 是    | String | 消息模板名称 |

表 6-554 Topics

| 参数           | 是否必选 | 参数类型    | 描述                                                  |
|--------------|------|---------|-----------------------------------------------------|
| name         | 是    | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | 是    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | 否    | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | 否    | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

## 响应参数

状态码：200

表 6-555 响应 Body 参数

| 参数                | 参数类型   | 描述                           |
|-------------------|--------|------------------------------|
| sql_alarm_rule_id | String | 告警规则id<br>最小长度：36<br>最大长度：36 |

状态码：400

表 6-556 响应 Body 参数

| 参数         | 参数类型   | 描述   |
|------------|--------|------|
| error_code | String | 错误码。 |

| 参数        | 参数类型   | 描述                                                                                         |
|-----------|--------|--------------------------------------------------------------------------------------------|
| error_msg | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

状态码：500

表 6-557 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                         |
|------------|--------|--------------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                       |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值：<br><ul style="list-style-type: none"> <li>Invalid projectId</li> </ul> |

## 请求示例

### 创建SQL告警规则

POST https://{endpoint}/v2/{project\_id}/lts/alarms/sql-alarm-rule

```
{
 "sql_alarm_rule_name": "huawei",
 "is_css_sql": false,
 "sql_alarm_rule_description": "huawei",
 "sql_requests": [{
 "log_stream_id": "1",
 "log_group_id": "1",
 "sql": "select count(*) as t",
 "sql_request_title": "demo",
 "search_time_range": 10,
 "search_time_range_unit": "minute"
 }],
 "frequency": {
 "type": "FIXED_RATE",
 "cron_expr": "",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 10,
 "fixed_rate_unit": "minute"
 },
 "condition_expression": "t>0",
 "sql_alarm_level": "Critical",
 "sql_alarm_send": true,
 "domain_id": "",
 "notification_frequency": 30,
 "alarm_action_rule_name": "",
 "notification_save_rule": {
 "language": "zh-cn",
 "timezone": "Asia/Shanghai",
 "user_name": "huawei",
 "template_name": "消息模板名称",
 "topics": [{
 "name": "huawei",
```

```
"topic_urn" : "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name" : "",
 "push_policy" : 0
}]
}
```

## 响应示例

### 状态码：200

请求响应成功。

```
{
 "sql_alarm_rule_id" : ""
}
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.2005",
 "error_msg" : "Alarm rule params validator error."
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2001",
 "error_msg" : "Failed to create alarm rule."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 创建SQL告警规则

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateSqlAlarmRuleSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
CreateSqlAlarmRuleRequest request = new CreateSqlAlarmRuleRequest();
CreateSqlAlarmRuleRequestBody body = new CreateSqlAlarmRuleRequestBody();
List<Topics> listNotificationSaveRuleTopics = new ArrayList<>();
listNotificationSaveRuleTopics.add(
 new Topics()
 .withName("huawei")
 .withTopicUrn("urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei")
 .withDisplayName("")
 .withPushPolicy(0)
);
SqlNotificationSaveRule notificationSaveRulebody = new SqlNotificationSaveRule();
notificationSaveRulebody.withLanguage(SqlNotificationSaveRule.LanguageEnum.fromValue("zh-cn"))
 .withTimezone("Asia/Shanghai")
 .withUserName("huawei")
 .withTopics(listNotificationSaveRuleTopics)
 .withTemplateName("消息模板名称");
CreateSqlAlarmRuleFrequency frequencybody = new CreateSqlAlarmRuleFrequency();
frequencybody.withType(CreateSqlAlarmRuleFrequency.TypeEnum.fromValue("FIXED_RATE"))
 .withCronExpr("")
 .withHourOfDay(0)
 .withDayOfWeek(0)
 .withFixedRate(10)
 .withFixedRateUnit(CreateSqlAlarmRuleFrequency.FixedRateUnitEnum.fromValue("minute"));
List<SqlRequest> listbodySqlRequests = new ArrayList<>();
listbodySqlRequests.add(
 new SqlRequest()
 .withLogStreamId("1")
 .withLogGroupId("1")
 .withSql("select count(*) as t")
 .withSqlRequestTitle("demo")
 .withSearchTimeRange(10)
 .withSearchTimeRangeUnit(SqlRequest.SearchTimeRangeUnitEnum.fromValue("minute"))
);
body.withAlarmActionRuleName("");

body.withNotificationFrequency(CreateSqlAlarmRuleRequestBody.NotificationFrequencyEnum.NUMBER_30);
body.withNotificationSaveRule(notificationSaveRulebody);
body.withDomainId("");
body.withSqlAlarmSend(true);
body.withSqlAlarmLevel(CreateSqlAlarmRuleRequestBody.SqlAlarmLevelEnum.fromValue("Critical"));
body.withConditionExpression(">0");
body.withFrequency(frequencybody);
body.withSqlRequests(listbodySqlRequests);
body.withSqlAlarmRuleDescription("huawei");
body.withIsCssSql(false);
body.withSqlAlarmRuleName("huawei");
request.withBody(body);
try {
 CreateSqlAlarmRuleResponse response = client.createSqlAlarmRule(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
}
```

```
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 创建SQL告警规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateSqlAlarmRuleRequest()
 listTopicsNotificationSaveRule = [
 Topics(
 name="huawei",
 topic_urn="urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 display_name="",
 push_policy=0
)
]
 notificationSaveRulebody = SqlNotificationSaveRule(
 language="zh-cn",
 timezone="Asia/Shanghai",
 user_name="huawei",
 topics=listTopicsNotificationSaveRule,
 template_name="消息模板名称"
)
 frequencybody = CreateSqlAlarmRuleFrequency(
 type="FIXED_RATE",
 cron_expr="",
 hour_of_day=0,
 day_of_week=0,
 fixed_rate=10,
 fixed_rate_unit="minute"
)
 listSqlRequestsbody = [
 SqlRequest(
 log_stream_id="1",
 log_group_id="1",
 sql="select count(*) as t",
 sql_request_title="demo",
 search_time_range=10,
 search_time_range_unit="minute"
)
]
```

```
)
]
request.body = CreateSqlAlarmRuleRequestBody(
 alarm_action_rule_name="",
 notification_frequency=30,
 notification_save_rule=notificationSaveRulebody,
 domain_id="",
 sql_alarm_send=True,
 sql_alarm_level="Critical",
 condition_expression="t>0",
 frequency=frequencybody,
 sql_requests=listSqlRequestsbody,
 sql_alarm_rule_description="huawei",
 is_css_sql=False,
 sql_alarm_rule_name="huawei"
)
response = client.create_sql_alarm_rule(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建SQL告警规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateSqlAlarmRuleRequest{
 displayNameTopics:= ""
 pushPolicyTopics:= int32(0)
 var listTopicsNotificationSaveRule = []model.Topics{
 {
 Name: "huawei",
 TopicUrn: "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 DisplayName: &displayNameTopics,
 PushPolicy: &pushPolicyTopics,
 }
 }
 }
}
```

```
 },
 }
 timezoneNotificationSaveRule:= "Asia/Shanghai"
 notificationSaveRulebody := &model.SqlNotificationSaveRule{
 Language: model.GetSqlNotificationSaveRuleLanguageEnum().ZH_CN,
 Timezone: &timezoneNotificationSaveRule,
 UserName: "huawei",
 Topics: listTopicsNotificationSaveRule,
 TemplateName: "消息模板名称",
 }
 cronExprFrequency:= ""
 hourOfDayFrequency:= int32(0)
 dayOfWeekFrequency:= int32(0)
 fixedRateFrequency:= int32(10)
 fixedRateUnitFrequency:= model.GetCreateSqlAlarmRuleFrequencyFixedRateUnitEnum().MINUTE
 frequencybody := &model.CreateSqlAlarmRuleFrequency{
 Type: model.GetCreateSqlAlarmRuleFrequencyTypeEnum().FIXED_RATE,
 CronExpr: &cronExprFrequency,
 HourOfDay: &hourOfDayFrequency,
 DayOfWeek: &dayOfWeekFrequency,
 FixedRate: &fixedRateFrequency,
 FixedRateUnit: &fixedRateUnitFrequency,
 }
 var listSqlRequestsbody = []model.SqlRequest{
 {
 LogStreamId: "1",
 LogGroupid: "1",
 Sql: "select count(*) as t",
 SqlRequestTitle: "demo",
 SearchTimeRange: int32(10),
 SearchTimeRangeUnit: model.GetSqlRequestSearchTimeRangeUnitEnum().MINUTE,
 },
 }
 alarmActionRuleNameCreateSqlAlarmRuleRequestBody:= ""
 sqlAlarmRuleDescriptionCreateSqlAlarmRuleRequestBody:= "huawei"
 isCssSqlCreateSqlAlarmRuleRequestBody:= false
 request.Body = &model.CreateSqlAlarmRuleRequestBody{
 AlarmActionRuleName: &alarmActionRuleNameCreateSqlAlarmRuleRequestBody,
 NotificationFrequency: model.GetCreateSqlAlarmRuleRequestBodyNotificationFrequencyEnum().E_30,
 NotificationSaveRule: notificationSaveRulebody,
 DomainId: "",
 SqlAlarmSend: true,
 SqlAlarmLevel: model.GetCreateSqlAlarmRuleRequestBodySqlAlarmLevelEnum().CRITICAL,
 ConditionExpression: "t>0",
 Frequency: frequencybody,
 SqlRequests: listSqlRequestsbody,
 SqlAlarmRuleDescription: &sqlAlarmRuleDescriptionCreateSqlAlarmRuleRequestBody,
 IsCssSql: &isCssSqlCreateSqlAlarmRuleRequestBody,
 SqlAlarmRuleName: "huawei",
 }
 response, err := client.CreateSqlAlarmRule(request)
 if err == nil {
 fmt.Printf("%v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.12.2 修改 SQL 告警规则

### 功能介绍

该接口用于修改SQL告警

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v2/{project\_id}/lts/alarms/sql-alarm-rule

表 6-558 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-559 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                                |
|--------------|------|--------|---------------------------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，<br>获取方式请参见： <a href="#">获取用户Token</a><br>最小长度： <b>1000</b><br>最大长度： <b>2000</b> |
| Content-Type | 是    | String | 该字段填为：application/<br>json;charset=UTF-8。<br>最小长度： <b>30</b><br>最大长度： <b>30</b>                   |

表 6-560 请求 Body 参数

| 参数                         | 是否必选 | 参数类型                                               | 描述                                                              |
|----------------------------|------|----------------------------------------------------|-----------------------------------------------------------------|
| sql_alarm_rule_id          | 是    | String                                             | SQL告警id<br>最小长度： <b>36</b><br>最大长度： <b>36</b>                   |
| sql_alarm_rule_name        | 是    | String                                             | 规则原始名称（不支持修改首次<br>创建的原始名称）<br>最小长度： <b>1</b><br>最大长度： <b>64</b> |
| alarm_rule_alias           | 否    | String                                             | 规则名称<br>最小长度： <b>1</b><br>最大长度： <b>64</b>                       |
| is_css_sql                 | 否    | Boolean                                            | 是否管道符sql查询。<br>缺省值： <b>false</b>                                |
| sql_alarm_rule_description | 否    | String                                             | SQL告警信息描述<br>最小长度： <b>0</b><br>最大长度： <b>64</b>                  |
| sql_requests               | 是    | Array of <a href="#">SqlRequest</a> objects        | SQL详细信息                                                         |
| frequency                  | 是    | <a href="#">CreateSqlAlarmRuleFrequency</a> object | 告警统计周期                                                          |

| 参数                          | 是否必选 | 参数类型                           | 描述                                                                                                                                         |
|-----------------------------|------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| condition_expression        | 是    | String                         | 条件表达式<br>最小长度：1<br>最大长度：2048                                                                                                               |
| sql_alarm_level             | 是    | String                         | 告警级别<br>枚举值： <ul style="list-style-type: none"><li>• Info</li><li>• Minor</li><li>• Major</li><li>• Critical</li></ul>                     |
| sql_alarm_send              | 是    | Boolean                        | 是否发送                                                                                                                                       |
| sql_alarm_send_code         | 是    | Integer                        | 发送主题 0:不变 1:新增 2:修改 3:删除<br>最小值：0<br>最大值：3<br>枚举值： <ul style="list-style-type: none"><li>• 0</li><li>• 1</li><li>• 2</li><li>• 3</li></ul> |
| domain_id                   | 是    | String                         | domainId<br>最小长度：32<br>最大长度：32                                                                                                             |
| notification_save_rule      | 否    | SqlNotificationSaveRule object | 通知主题                                                                                                                                       |
| trigger_condition_count     | 否    | Integer                        | 触发条件：触发次数；默认为1                                                                                                                             |
| trigger_condition_frequency | 否    | Integer                        | 触发条件：触发周期；默认为1                                                                                                                             |
| whether_recovery_policy     | 否    | Boolean                        | 是否打开恢复通知；默认false                                                                                                                           |
| recovery_policy             | 否    | Integer                        | 恢复策略周期；默认为3                                                                                                                                |

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                                                                           |
|------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| notification_frequency | 是    | Integer | 通知频率,单位(分钟)<br>枚举值: <ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name | 否    | String  | 告警行动规则名称<br><b>说明</b><br>alarm_action_rule_name和notification_save_rule可以选填一个, 如果都填, 优先选择alarm_action_rule_name                                                               |

表 6-561 SqlRequest

| 参数                     | 是否必选 | 参数类型    | 描述                                           |
|------------------------|------|---------|----------------------------------------------|
| is_time_range_relative | 否    | Boolean | 是时间范围相对                                      |
| log_stream_id          | 是    | String  | 日志流id<br>最小长度: <b>36</b><br>最大长度: <b>36</b>  |
| log_stream_name        | 否    | String  | 日志流名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>   |
| log_group_id           | 是    | String  | 日志组id<br>最小长度: <b>36</b><br>最大长度: <b>36</b>  |
| log_group_name         | 否    | String  | 日志组名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>   |
| sql                    | 是    | String  | sql语句<br>最小长度: <b>1</b><br>最大长度: <b>1024</b> |

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                     |
|------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------|
| sql_request_title      | 是    | String  | 图表名称<br>最小长度：1<br>最大长度：64                                                                                              |
| search_time_range      | 是    | Integer | 查询执行任务时最近数据的时间范围（当 search_time_range_unit 为 minute，则最大值为60；当 search_time_range_unit 为 hour，则最大值为24）<br>最小值：1<br>最大值：60 |
| search_time_range_unit | 是    | String  | 查询时间单位<br>枚举值：<br><ul style="list-style-type: none"> <li>minute</li> <li>hour</li> </ul>                               |

表 6-562 CreateSqlAlarmRuleFrequency

| 参数          | 是否必选 | 参数类型    | 描述                                                                                                                                         |
|-------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------|
| type        | 是    | String  | 时间类型。<br>枚举值：<br><ul style="list-style-type: none"> <li>CRON</li> <li>HOURLY</li> <li>DAILY</li> <li>WEEKLY</li> <li>FIXED_RATE</li> </ul> |
| cron_expr   | 否    | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                |
| hour_of_day | 否    | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。<br>DAILY：最小值：0，最大值：23<br>WEEKLY：最小值：0，最大值：23                                                               |
| day_of_week | 否    | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。                                                                                                              |

| 参数              | 是否必选 | 参数类型    | 描述                                                                                                           |
|-----------------|------|---------|--------------------------------------------------------------------------------------------------------------|
| fixed_rate      | 否    | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24）。<br>最小值：1<br>最大值：60 |
| fixed_rate_unit | 否    | String  | 时间单位。<br>枚举值： <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul>                        |

表 6-563 SqlNotificationSaveRule

| 参数            | 是否必选 | 参数类型                           | 描述                                                                                                            |
|---------------|------|--------------------------------|---------------------------------------------------------------------------------------------------------------|
| language      | 是    | String                         | 首选项对应的语言<br>最小长度：0<br>最大长度：10<br>枚举值： <ul style="list-style-type: none"><li>• zh-cn</li><li>• en-us</li></ul> |
| timezone      | 否    | String                         | 首选项对应的时区信息<br>最小长度：0<br>最大长度：1024                                                                             |
| user_name     | 是    | String                         | 用户名<br>最小长度：1<br>最大长度：1024                                                                                    |
| topics        | 是    | Array of <b>Topics</b> objects | 主题信息                                                                                                          |
| template_name | 是    | String                         | 消息模板名称                                                                                                        |

表 6-564 Topics

| 参数           | 是否必选 | 参数类型    | 描述                                                  |
|--------------|------|---------|-----------------------------------------------------|
| name         | 是    | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | 是    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | 否    | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | 否    | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

## 响应参数

状态码：200

表 6-565 响应 Body 参数

| 参数                  | 参数类型    | 描述                               |
|---------------------|---------|----------------------------------|
| sql_alarm_rule_name | String  | 原始规则名称                           |
| alarm_rule_alias    | String  | 规则名称<br>最小长度：1<br>最大长度：64        |
| is_css_sql          | Boolean | 是否管道符sql查询。<br>缺省值： <b>false</b> |
| indexId             | String  | 索引id                             |
| projectId           | String  | 项目id                             |
| sql_alarm_rule_id   | String  | SQL告警规则id<br>最小长度：36<br>最大长度：36  |

| 参数                         | 参数类型                                        | 描述                                                                                                                     |
|----------------------------|---------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| sql_alarm_rule_description | String                                      | SQL告警信息描述<br>最小长度：0<br>最大长度：64                                                                                         |
| sql_requests               | Array of <a href="#">SqlRequest</a> objects | SQL详细信息                                                                                                                |
| frequency                  | <a href="#">FrequencyRespBody</a> object    | 告警统计周期                                                                                                                 |
| condition_expression       | String                                      | 条件表达式<br>最小长度：1<br>最大长度：1024                                                                                           |
| sql_alarm_level            | String                                      | 告警级别<br>枚举值： <ul style="list-style-type: none"><li>• Info</li><li>• Minor</li><li>• Major</li><li>• CRITICAL</li></ul> |
| sql_alarm_send             | Boolean                                     | 是否发送                                                                                                                   |
| domain_id                  | String                                      | domainId<br>最小长度：32<br>最大长度：32                                                                                         |
| create_time                | Long                                        | 创建时间（毫秒时间戳）<br>最小值：13<br>最大值：13                                                                                        |
| update_time                | Long                                        | 更新时间（毫秒时间戳）<br>最小值：13<br>最大值：13                                                                                        |
| topics                     | Array of <a href="#">Topics</a> objects     | 主题                                                                                                                     |
| language                   | String                                      | 邮件附加信息语言<br>枚举值： <ul style="list-style-type: none"><li>• zh-cn</li><li>• en-us</li></ul>                               |
| id                         | String                                      | 规则ID。                                                                                                                  |

| 参数                     | 参数类型    | 描述                                                                                                                                                                           |
|------------------------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| notification_frequency | Integer | 通知频率,单位(分钟)<br>枚举值: <ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name | String  | 告警行动规则名称<br>说明<br>alarm_action_rule_name和notification_save_rule可以选填一个, 如果都填, 优先选择alarm_action_rule_name                                                                      |

表 6-566 SqlRequest

| 参数                     | 参数类型    | 描述                             |
|------------------------|---------|--------------------------------|
| is_time_range_relative | Boolean | 是时间范围相对                        |
| log_stream_id          | String  | 日志流id<br>最小长度: 36<br>最大长度: 36  |
| log_stream_name        | String  | 日志流名称<br>最小长度: 1<br>最大长度: 64   |
| log_group_id           | String  | 日志组id<br>最小长度: 36<br>最大长度: 36  |
| log_group_name         | String  | 日志组名称<br>最小长度: 1<br>最大长度: 64   |
| sql                    | String  | sql语句<br>最小长度: 1<br>最大长度: 1024 |

| 参数                     | 参数类型    | 描述                                                                                                               |
|------------------------|---------|------------------------------------------------------------------------------------------------------------------|
| sql_request_title      | String  | 图表名称<br>最小长度：1<br>最大长度：64                                                                                        |
| search_time_range      | Integer | 查询执行任务时最近数据的时间范围（当search_time_range_unit为minute，则最大值为60；当search_time_range_unit为hour，则最大值为24）<br>最小值：1<br>最大值：60 |
| search_time_range_unit | String  | 查询时间单位<br>枚举值： <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul>                           |

表 6-567 FrequencyResponseBody

| 参数          | 参数类型    | 描述                                                                                                                                          |
|-------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------|
| type        | String  | 时间类型。<br>枚举值： <ul style="list-style-type: none"><li>• CRON</li><li>• HOURLY</li><li>• DAILY</li><li>• WEEKLY</li><li>• FIXED_RATE</li></ul> |
| cron_expr   | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                 |
| hour_of_day | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。                                                                                                             |
| day_of_week | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。                                                                                                               |
| fixed_rate  | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24）。                                                   |

| 参数              | 参数类型   | 描述                                                                                       |
|-----------------|--------|------------------------------------------------------------------------------------------|
| fixed_rate_unit | String | 时间单位枚举值：<br>枚举值： <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul> |

表 6-568 Topics

| 参数           | 参数类型    | 描述                                                  |
|--------------|---------|-----------------------------------------------------|
| name         | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

状态码：400

表 6-569 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                      |
|------------|--------|-----------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                    |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>• Invalid projectId</li></ul> |

状态码：500

表 6-570 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

### 修改SQL告警规则

PUT https://{endpoint}/v2/{project\_id}/lts/alarms/sql-alarm-rule

```
{
 "sql_alarm_rule_id": "",
 "sql_alarm_rule_name": "huawei",
 "is_css_sql": false,
 "alarm_rule_alias": "zhangsan",
 "sql_alarm_rule_description": "huawei",
 "sql_requests": [{
 "log_stream_id": "1",
 "log_group_id": "1",
 "sql": "select count(*) as t",
 "sql_request_title": "demo",
 "search_time_range": 10,
 "search_time_range_unit": "minute"
 }],
 "frequency": {
 "type": "FIXED_RATE",
 "cron_expr": "",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 10,
 "fixed_rate_unit": "minute"
 },
 "condition_expression": "t>0",
 "sql_alarm_level": "Critical",
 "sql_alarm_send": true,
 "sql_alarm_send_code": 0,
 "domain_id": "",
 "notification_frequency": 5,
 "alarm_action_rule_name": "",
 "notification_save_rule": {
 "language": "zh-cn",
 "timezone": "Asia/Shanghai",
 "user_name": "huawei",
 "template_name": "消息模板名称",
 "topics": [{
 "name": "huawei",
 "topic_urn": "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name": "",
 "push_policy": 0
 }]
 }
}
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "sql_alarm_rule_id": "",
 "sql_alarm_rule_name": "huawei",
 "alarm_rule_alias": "zhangsan",
 "is_css_sql": false,
 "sql_alarm_rule_description": "huawei",
 "sql_requests": [{
 "log_stream_id": "1",
 "log_stream_name": "huawei",
 "log_group_name": "huawei",
 "log_group_id": "1",
 "sql": "select count(*) as t",
 "sql_request_title": "demo",
 "search_time_range": 10,
 "search_time_range_unit": "minute"
 }],
 "frequency": {
 "type": "FIXED_RATE",
 "cron_expr": "",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 10,
 "fixed_rate_unit": "minute"
 },
 "condition_expression": "t>0",
 "sql_alarm_level": "Critical",
 "sql_alarm_send": true,
 "domain_id": "",
 "notification_frequency": 5,
 "alarm_action_rule_name": "",
 "topics": [{
 "name": "huawei",
 "topic_urn": "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name": "",
 "push_policy": 0
 }]
}
```

#### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.2005",
 "error_msg": "Alarm rule params validator error."
}
```

#### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2003",
 "error_msg": "Failed to update alarm rule."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

修改SQL告警规则

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateSqlAlarmRuleSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateSqlAlarmRuleRequest request = new UpdateSqlAlarmRuleRequest();
 UpdateSqlAlarmRuleRequestBody body = new UpdateSqlAlarmRuleRequestBody();
 List<Topics> listNotificationSaveRuleTopics = new ArrayList<>();
 listNotificationSaveRuleTopics.add(
 new Topics()
 .withName("huawei")
 .withTopicUrn("urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei")
 .withDisplayName("")
 .withPushPolicy(0)
);
 SqlNotificationSaveRule notificationSaveRulebody = new SqlNotificationSaveRule();
 notificationSaveRulebody.withLanguage(SqlNotificationSaveRule.LanguageEnum.fromValue("zh-cn"))
 .withTimezone("Asia/Shanghai")
 .withUserName("huawei")
 .withTopics(listNotificationSaveRuleTopics)
 .withTemplateName("消息模板名称");
 CreateSqlAlarmRuleFrequency frequencybody = new CreateSqlAlarmRuleFrequency();
 frequencybody.withType(CreateSqlAlarmRuleFrequency.TypeEnum.fromValue("FIXED_RATE"))
 .withCronExpr("")
 .withHourOfDay(0)
 .withDayOfWeek(0)
 .withFixedRate(10)
 .withFixedRateUnit(CreateSqlAlarmRuleFrequency.FixedRateUnitEnum.fromValue("minute"));
 List<SqlRequest> listbodySqlRequests = new ArrayList<>();
 listbodySqlRequests.add(
 new SqlRequest()
 .withLogStreamId("1")
 .withLogGroupId("1")
 .withSql("select count(*) as t")
 .withSqlRequestTitle("demo")
 .withSearchTimeRange(10)
 .withSearchTimeRangeUnit(SqlRequest.SearchTimeRangeUnitEnum.fromValue("minute"))
);
 body.withAlarmActionRuleName("");

 body.withNotificationFrequency(UpdateSqlAlarmRuleRequestBody.NotificationFrequencyEnum.NUMBER_5);
 }
}
```

```
body.withNotificationSaveRule(notificationSaveRulebody);
body.withDomainId("");
body.withSqlAlarmSendCode(UpdateSqlAlarmRuleRequestBody.SqlAlarmSendCodeEnum.NUMBER_0);
body.withSqlAlarmSend(true);
body.withSqlAlarmLevel(UpdateSqlAlarmRuleRequestBody.SqlAlarmLevelEnum.fromValue("Critical"));
body.withConditionExpression(">0");
body.withFrequency(frequencybody);
body.withSqlRequests(listbodySqlRequests);
body.withSqlAlarmRuleDescription("huawei");
body.withIsCssSql(false);
body.withAlarmRuleAlias("zhangsan");
body.withSqlAlarmRuleName("huawei");
body.withSqlAlarmRuleId("");
request.withBody(body);
try {
 UpdateSqlAlarmRuleResponse response = client.updateSqlAlarmRule(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 修改SQL告警规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateSqlAlarmRuleRequest()
 listTopicsNotificationSaveRule = [
 Topics(
 name="huawei",
 topic_urn="urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 display_name="",
 push_policy=0
)
]
```

```
]
notificationSaveRulebody = SqlNotificationSaveRule(
 language="zh-cn",
 timezone="Asia/Shanghai",
 user_name="huawei",
 topics=listTopicsNotificationSaveRule,
 template_name="消息模板名称"
)
frequencybody = CreateSqlAlarmRuleFrequency(
 type="FIXED_RATE",
 cron_expr="",
 hour_of_day=0,
 day_of_week=0,
 fixed_rate=10,
 fixed_rate_unit="minute"
)
listSqlRequestsbody = [
 SqlRequest(
 log_stream_id="1",
 log_group_id="1",
 sql="select count(*) as t",
 sql_request_title="demo",
 search_time_range=10,
 search_time_range_unit="minute"
)
]
request.body = UpdateSqlAlarmRuleRequestBody(
 alarm_action_rule_name="",
 notification_frequency=5,
 notification_save_rule=notificationSaveRulebody,
 domain_id="",
 sql_alarm_send_code=0,
 sql_alarm_send=True,
 sql_alarm_level="Critical",
 condition_expression="t>0",
 frequency=frequencybody,
 sql_requests=listSqlRequestsbody,
 sql_alarm_rule_description="huawei",
 is_css_sql=False,
 alarm_rule_alias="zhangsan",
 sql_alarm_rule_name="huawei",
 sql_alarm_rule_id=""
)
response = client.update_sql_alarm_rule(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 修改SQL告警规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.UpdateSqlAlarmRuleRequest{
 displayNameTopics:= ""
 pushPolicyTopics:= int32(0)
 var listTopicsNotificationSaveRule = []model.Topics{
 {
 Name: "huawei",
 TopicUrn: "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 DisplayName: &displayNameTopics,
 PushPolicy: &pushPolicyTopics,
 },
 },
 timezoneNotificationSaveRule:= "Asia/Shanghai"
 notificationSaveRulebody := &model.SqlNotificationSaveRule{
 Language: model.GetSqlNotificationSaveRuleLanguageEnum().ZH_CN,
 Timezone: &timezoneNotificationSaveRule,
 UserName: "huawei",
 Topics: listTopicsNotificationSaveRule,
 TemplateName: "消息模板名称",
 }
 cronExprFrequency:= ""
 hourOfDayFrequency:= int32(0)
 dayOfWeekFrequency:= int32(0)
 fixedRateFrequency:= int32(10)
 fixedRateUnitFrequency:= model.GetCreateSqlAlarmRuleFrequencyFixedRateUnitEnum().MINUTE
 frequencybody := &model.CreateSqlAlarmRuleFrequency{
 Type: model.GetCreateSqlAlarmRuleFrequencyTypeEnum().FIXED_RATE,
 CronExpr: &cronExprFrequency,
 HourOfDay: &hourOfDayFrequency,
 DayOfWeek: &dayOfWeekFrequency,
 FixedRate: &fixedRateFrequency,
 FixedRateUnit: &fixedRateUnitFrequency,
 }
 var listSqlRequestsbody = []model.SqlRequest{
 {
 LogStreamId: "1",
 LogGroupID: "1",
 Sql: "select count(*) as t",
 SqlRequestTitle: "demo",
 SearchTimeRange: int32(10),
 SearchTimeRangeUnit: model.GetSqlRequestSearchTimeRangeUnitEnum().MINUTE,
 },
 },
 }
 alarmActionRuleNameUpdateSqlAlarmRuleRequestBody:= ""
 sqlAlarmRuleDescriptionUpdateSqlAlarmRuleRequestBody:= "huawei"
 isCssSqlUpdateSqlAlarmRuleRequestBody:= false
 alarmRuleAliasUpdateSqlAlarmRuleRequestBody:= "zhongsan"
 request.Body = &model.UpdateSqlAlarmRuleRequestBody{
 AlarmActionRuleName: &alarmActionRuleNameUpdateSqlAlarmRuleRequestBody,
 NotificationFrequency: model.GetUpdateSqlAlarmRuleRequestBodyNotificationFrequencyEnum().E_5,
 NotificationSaveRule: notificationSaveRulebody,
 DomainId: "",
 }
}
```

```
SqlAlarmSendCode: model.GetUpdateSqlAlarmRuleRequestBodySqlAlarmSendCodeEnum().E_0,
SqlAlarmSend: true,
SqlAlarmLevel: model.GetUpdateSqlAlarmRuleRequestBodySqlAlarmLevelEnum().CRITICAL,
ConditionExpression: "t>0",
Frequency: frequencybody,
SqlRequests: listSqlRequestsbody,
SqlAlarmRuleDescription: &sqlAlarmRuleDescriptionUpdateSqlAlarmRuleRequestBody,
IsCssSql: &isCssSqlUpdateSqlAlarmRuleRequestBody,
AlarmRuleAlias: &alarmRuleAliasUpdateSqlAlarmRuleRequestBody,
SqlAlarmRuleName: "huawei",
SqlAlarmRuleId: "",
}
response, err := client.UpdateSqlAlarmRule(request)
if err == nil {
 fmt.Printf("%v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.12.3 查询 SQL 告警规则

### 功能介绍

该接口用于查询SQL告警

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/lts/alarms/sql-alarm-rule

表 6-571 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-572 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

## 响应参数

状态码: 200

表 6-573 响应 Body 参数

| 参数              | 参数类型                                                   | 描述    |
|-----------------|--------------------------------------------------------|-------|
| sql_alarm_rules | Array of <a href="#">SqlAlarmRuleResplList</a> objects | SQL告警 |

表 6-574 SqlAlarmRuleRespList

| 参数                         | 参数类型                                        | 描述                                                                                                                     |
|----------------------------|---------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| sql_alarm_rule_name        | String                                      | SQL告警名称<br>最小长度：1<br>最大长度：64                                                                                           |
| is_css_sql                 | Boolean                                     | 是否管道符sql查询。<br>缺省值：false                                                                                               |
| sql_alarm_rule_id          | String                                      | SQL告警规则id<br>最小长度：36<br>最大长度：36                                                                                        |
| sql_alarm_rule_description | String                                      | SQL告警信息描述<br>最小长度：0<br>最大长度：64                                                                                         |
| sql_requests               | Array of <a href="#">SqlRequest</a> objects | SQL详细信息                                                                                                                |
| frequency                  | <a href="#">FrequencyRespBody</a> object    | 告警统计周期                                                                                                                 |
| condition_expression       | String                                      | 条件表达式<br>最小长度：1<br>最大长度：1024                                                                                           |
| topics                     | Array of <a href="#">Topics</a> objects     | 主题信息                                                                                                                   |
| sql_alarm_level            | String                                      | 告警级别<br>枚举值： <ul style="list-style-type: none"><li>• Info</li><li>• Minor</li><li>• Major</li><li>• Critical</li></ul> |
| sql_alarm_send             | Boolean                                     | 是否发送                                                                                                                   |
| domain_id                  | String                                      | domainId<br>最小长度：32<br>最大长度：32                                                                                         |
| create_time                | Long                                        | 创建时间（毫秒时间戳）<br>最小值：13<br>最大值：13                                                                                        |

| 参数                          | 参数类型    | 描述                                                                                                                                                                              |
|-----------------------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| update_time                 | Long    | 更新时间（毫秒时间戳）<br>最小值：13<br>最大值：13                                                                                                                                                 |
| template_name               | String  | 消息模板名称                                                                                                                                                                          |
| status                      | String  | 告警状态<br>枚举值：<br><ul style="list-style-type: none"> <li>• <b>RUNNING</b> 启用</li> <li>• <b>STOPPING</b> 停止</li> </ul>                                                             |
| trigger_condition_count     | Integer | 触发条件：触发周期；默认为1                                                                                                                                                                  |
| trigger_condition_frequency | Integer | 触发条件：触发周期；默认为1                                                                                                                                                                  |
| whether_recovery_policy     | Boolean | 是否打开恢复通知；默认false                                                                                                                                                                |
| recovery_policy             | Integer | 恢复策略周期；默认为3                                                                                                                                                                     |
| notification_frequency      | Integer | 通知频率,单位(分钟)<br>枚举值：<br><ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name      | String  | 告警行动规则名称<br>说明<br>alarm_action_rule_name和notification_save_rule可以选填一个，如果都填，优先选择alarm_action_rule_name                                                                           |

表 6-575 SqlRequest

| 参数                     | 参数类型    | 描述      |
|------------------------|---------|---------|
| is_time_range_relative | Boolean | 是时间范围相对 |

| 参数                     | 参数类型    | 描述                                                                                                                        |
|------------------------|---------|---------------------------------------------------------------------------------------------------------------------------|
| log_stream_id          | String  | 日志流id<br>最小长度: 36<br>最大长度: 36                                                                                             |
| log_stream_name        | String  | 日志流名称<br>最小长度: 1<br>最大长度: 64                                                                                              |
| log_group_id           | String  | 日志组id<br>最小长度: 36<br>最大长度: 36                                                                                             |
| log_group_name         | String  | 日志组名称<br>最小长度: 1<br>最大长度: 64                                                                                              |
| sql                    | String  | sql语句<br>最小长度: 1<br>最大长度: 1024                                                                                            |
| sql_request_title      | String  | 图表名称<br>最小长度: 1<br>最大长度: 64                                                                                               |
| search_time_range      | Integer | 查询执行任务时最近数据的时间范围<br>(当search_time_range_unit为minute, 则最大值为60; 当search_time_range_unit为hour, 则最大值为24)<br>最小值: 1<br>最大值: 60 |
| search_time_range_unit | String  | 查询时间单位<br>枚举值: <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul>                                    |

表 6-576 FrequencyRespBody

| 参数              | 参数类型    | 描述                                                                                                                                          |
|-----------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------|
| type            | String  | 时间类型。<br>枚举值： <ul style="list-style-type: none"><li>• CRON</li><li>• HOURLY</li><li>• DAILY</li><li>• WEEKLY</li><li>• FIXED_RATE</li></ul> |
| cron_expr       | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                 |
| hour_of_day     | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。                                                                                                             |
| day_of_week     | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。                                                                                                               |
| fixed_rate      | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24）。                                                   |
| fixed_rate_unit | String  | 时间单位枚举值：<br>枚举值： <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul>                                                    |

表 6-577 Topics

| 参数           | 参数类型   | 描述                                                  |
|--------------|--------|-----------------------------------------------------|
| name         | String | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | String | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | String | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |

| 参数          | 参数类型    | 描述                            |
|-------------|---------|-------------------------------|
| push_policy | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024 |

状态码：500

表 6-578 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

查询SQL告警规则

```
GET https://{endpoint}/v2/{project_id}/lts/alarms/sql-alarm-rule
/v2/{project_id}/lts/alarms/sql-alarm-rule
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "sql_alarm_rules": [{
 "sql_alarm_rule_name": "string",
 "is_css_sql": false,
 "sql_alarm_rule_id": "string",
 "sql_alarm_rule_description": "string",
 "sql_requests": [{
 "log_stream_id": "string",
 "log_stream_name": "string",
 "log_group_id": "string",
 "log_group_name": "string",
 "sql": "string",
 "sql_request_title": "string",
 "search_time_range": 0,
 "search_time_range_unit": "minute"
 }],
 "frequency": {
 "type": "CRON",
 "cron_expr": "string",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 0,
 "fixed_rate_unit": "minute"
 }
 }],
}
```

```
"condition_expression" : "string",
"topics" : [{
 "name" : "string",
 "topic_urn" : "string",
 "display_name" : "string",
 "push_policy" : 0
}],
"sql_alarm_level" : "Info",
"sql_alarm_send" : true,
"domain_id" : "string",
"create_time" : 0,
"update_time" : 0,
"template_name" : "消息模板名称",
"status" : "RUNNING",
"trigger_condition_count" : "1",
"trigger_condition_frequency" : "1",
"whether_recovery_policy" : false,
"recovery_policy" : "3",
"notification_frequency" : 5,
"alarm_action_rule_name" : ""
}]
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2008",
 "error_msg" : "Find Alarm rule failed."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.12.4 删除 SQL 告警规则

### 功能介绍

该接口用于删除SQL告警

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v2/{project\_id}/lts/alarms/sql-alarm-rule/{sql\_alarm\_rule\_id}

表 6-579 路径参数

| 参数                | 是否必选 | 参数类型   | 描述                                                                                 |
|-------------------|------|--------|------------------------------------------------------------------------------------|
| project_id        | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |
| sql_alarm_rule_id | 是    | String | Sql告警规则id。<br>最小长度: 36<br>最大长度: 36                                                 |

## 请求参数

表 6-580 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

## 响应参数

状态码: 500

表 6-581 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值: <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

根据告警ID删除SQL告警规则

```
POST https://{endpoint}/v2/{project_id}/lts/alarms/sql-alarm-rule/{sql_alarm_rule_id}
/v2/{project_id}/lts/alarms/sql-alarm-rule/{sql_alarm_rule_id}
```

## 响应示例

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2002",
 "error_msg": "Failed to delete alarm rule."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.12.5 切换告警规则状态

### 功能介绍

该接口用于告警规则的启动与停止

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v2/{project\_id}/lts/alarms/status

表 6-582 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-583 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

表 6-584 请求 Body 参数

| 参数            | 是否必选 | 参数类型   | 描述                                              |
|---------------|------|--------|-------------------------------------------------|
| alarm_rule_id | 是    | String | 告警规则id。<br>最小长度: 1<br>最大长度: 100                 |
| type          | 是    | String | 告警规则类型“sql”或“keywords”。<br>最小长度: 1<br>最大长度: 100 |

| 参数     | 是否必选 | 参数类型   | 描述                                                                                                                                             |
|--------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------|
| status | 是    | String | RUNNING开启告警规则,STOPPING停止告警规则。<br>最小长度：1<br>最大长度：100<br>枚举值：<br><ul style="list-style-type: none"> <li>● RUNNING</li> <li>● STOPPING</li> </ul> |

## 响应参数

状态码：401

表 6-585 响应 Body 参数

| 参数         | 参数类型   | 描述                                 |
|------------|--------|------------------------------------|
| error_code | String | 错误码示例。<br>最小长度：10<br>最大长度：10       |
| error_msg  | String | 调用失败响应信息描述。<br>最小长度：1<br>最大长度：1000 |

状态码：500

表 6-586 响应 Body 参数

| 参数         | 参数类型   | 描述                                 |
|------------|--------|------------------------------------|
| error_code | String | 错误码示例。<br>最小长度：10<br>最大长度：10       |
| error_msg  | String | 调用失败响应信息描述。<br>最小长度：1<br>最大长度：1000 |

## 请求示例

- 启用指定告警规则ID的告警规则。“type”类型为：“sql”；“status”类型为：“RUNNING”。

```
PUT https://{endpoint}/v2/{project_id}/lts/alarms/status
```

```
{
 "alarm_rule_id": "ABCDEFGF",
 "type": "sql",
 "status": "RUNNING"
}
```

- “type” 类型为: "keywords"; "status"类型为: "STOPPING"。

```
PUT https://{endpoint}/v2/{project_id}/lts/alarms/status
```

```
{
 "alarm_rule_id": "ABCDEFGF",
 "type": "keywords",
 "status": "STOPPING"
}
```

## 响应示例

### 状态码: 401

项目id验证错误。

```
{
 "error_code": "LTS.0001",
 "error_msg": "Project verify error."
}
```

### 状态码: 500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2021",
 "error_msg": "Failed to update alarm rule status."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

- 启用指定告警规则ID的告警规则。“type” 类型为: "sql"; "status"类型为: "RUNNING"。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateAlarmRuleStatusSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
```

```
environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateAlarmRuleStatusRequest request = new UpdateAlarmRuleStatusRequest();
ChangeAlarmRuleStatus body = new ChangeAlarmRuleStatus();
body.withType("sql");
body.withStatus("RUNNING");
body.withAlarmRuleId("ABCDEFGH");
request.withBody(body);
try {
 UpdateAlarmRuleStatusResponse response = client.updateAlarmRuleStatus(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

- “type” 类型为: "keywords"; "status"类型为: "STOPPING"。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class UpdateAlarmRuleStatusSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
```

```
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
UpdateAlarmRuleStatusRequest request = new UpdateAlarmRuleStatusRequest();
ChangeAlarmRuleStatus body = new ChangeAlarmRuleStatus();
body.withType("keywords");
body.withStatus("STOPPING");
body.withAlarmRuleId("ABCDEFGH");
request.withBody(body);
try {
 UpdateAlarmRuleStatusResponse response = client.updateAlarmRuleStatus(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

- 启用指定告警规则ID的告警规则。“type”类型为：“sql”；“status”类型为：“RUNNING”。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 # environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 # environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateAlarmRuleStatusRequest()
 request.body = ChangeAlarmRuleStatus(
 type="sql",
 status="RUNNING",
 alarm_rule_id="ABCDEFGH"
)
 response = client.update_alarm_rule_status(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

- “type” 类型为: "keywords"; "status"类型为: "STOPPING"。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateAlarmRuleStatusRequest()
 request.body = ChangeAlarmRuleStatus(
 type="keywords",
 status="STOPPING",
 alarm_rule_id="ABCDEFGH"
)
 response = client.update_alarm_rule_status(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

- 启用指定告警规则ID的告警规则。“type” 类型为: "sql"; "status"类型为: "RUNNING"。

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
```

```
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.UpdateAlarmRuleStatusRequest{}
request.Body = &model.ChangeAlarmRuleStatus{
 Type: "sql",
 Status: "RUNNING",
 AlarmRuleId: "ABCDEFGH",
}
response, err := client.UpdateAlarmRuleStatus(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

- “type” 类型为: "keywords"; "status"类型为: "STOPPING"。

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before
 // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
 // environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateAlarmRuleStatusRequest{}
 request.Body = &model.ChangeAlarmRuleStatus{
 Type: "keywords",
 Status: "STOPPING",
 AlarmRuleId: "ABCDEFGH",
 }
 response, err := client.UpdateAlarmRuleStatus(request)
 if err == nil {
```

```
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 401 | 项目id验证错误。              |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

# 6.13 关键词告警规则

## 6.13.1 创建关键词告警规则

### 功能介绍

该接口用于创建关键词告警，目前每个账户最多可以创建共200个关键词告警与SQL告警。

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/lts/alarms/keywords-alarm-rule

表 6-587 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-588 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

表 6-589 请求 Body 参数

| 参数                              | 是否必选 | 参数类型                                             | 描述                                                              |
|---------------------------------|------|--------------------------------------------------|-----------------------------------------------------------------|
| keywords_alarm_rule_name        | 是    | String                                           | 关键词告警名称<br><b>说明</b><br>不能以点和下划线开头或以点结尾。<br>最小长度: 1<br>最大长度: 64 |
| keywords_alarm_rule_description | 否    | String                                           | 关键词告警信息描述<br>最小长度: 0<br>最大长度: 64                                |
| keywords_requests               | 是    | Array of <a href="#">KeywordsRequest</a> objects | 关键词详细信息                                                         |
| frequency                       | 是    | <a href="#">Frequency</a> object                 | 告警统计周期                                                          |

| 参数                          | 是否必选 | 参数类型                           | 描述                                                                                                                                                                              |
|-----------------------------|------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords_alarm_level        | 是    | String                         | 告警级别<br>枚举值：<br><ul style="list-style-type: none"> <li>• Info</li> <li>• Minor</li> <li>• Major</li> <li>• Critical</li> </ul>                                                  |
| keywords_alarm_send         | 是    | Boolean                        | 是否发送                                                                                                                                                                            |
| domain_id                   | 是    | String                         | domainId<br>最小长度：32<br>最大长度：32                                                                                                                                                  |
| notification_save_rule      | 否    | SqlNotificationSaveRule object | 通知主题                                                                                                                                                                            |
| trigger_condition_count     | 否    | Integer                        | 触发条件：触发次数；默认为1                                                                                                                                                                  |
| trigger_condition_frequency | 否    | Integer                        | 触发条件：触发周期；默认为1                                                                                                                                                                  |
| whether_recover_policy      | 否    | Boolean                        | 是否打开恢复通知；默认false                                                                                                                                                                |
| recovery_policy             | 否    | Integer                        | 恢复策略周期；默认为3                                                                                                                                                                     |
| notification_frequency      | 是    | Integer                        | 通知频率,单位(分钟)<br>枚举值：<br><ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name      | 否    | String                         | 告警行动规则名称<br><b>说明</b><br>alarm_action_rule_name和notification_save_rule可以选填一个，如果都填，优先选择alarm_action_rule_name                                                                    |

表 6-590 KeywordsRequest

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                       |
|------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------|
| log_stream_id          | 是    | String  | 日志流id<br>最小长度：36<br>最大长度：36                                                                                              |
| log_stream_name        | 否    | String  | 日志流名称<br>最小长度：1<br>最大长度：64                                                                                               |
| log_group_id           | 是    | String  | 日志组id<br>最小长度：36<br>最大长度：36                                                                                              |
| log_group_name         | 否    | String  | 日志组名称<br>最小长度：1<br>最大长度：64                                                                                               |
| keywords               | 是    | String  | 关键词<br>最小长度：1<br>最大长度：1024                                                                                               |
| condition              | 是    | String  | 条件<br>枚举值：<br><ul style="list-style-type: none"> <li>• &gt;=</li> <li>• &lt;=</li> <li>• &lt;</li> <li>• &gt;</li> </ul> |
| number                 | 是    | Integer | 行数<br>最小值：1<br>最大值：2147483647                                                                                            |
| search_time_range      | 是    | Integer | 查询执行任务时最近数据的时间范围，最大值为60 minute：最小值：1；最大值：60 hour：最小值：1；最大值：24                                                            |
| search_time_range_unit | 是    | String  | 查询时间单位<br>枚举值：<br><ul style="list-style-type: none"> <li>• minute</li> </ul>                                             |

表 6-591 Frequency

| 参数              | 是否必选 | 参数类型    | 描述                                                                                                                                                   |
|-----------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| type            | 是    | String  | 时间类型。<br>枚举值：<br><ul style="list-style-type: none"> <li>• CRON</li> <li>• HOURLY</li> <li>• DAILY</li> <li>• WEEKLY</li> <li>• FIXED_RATE</li> </ul> |
| cron_expr       | 否    | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                          |
| hour_of_day     | 否    | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。<br>DAILY：最小值：0，最大值：23<br>WEEKLY：最小值：0，最大值：23                                                                         |
| day_of_week     | 否    | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。<br>最小值：1<br>最大值：7                                                                                                      |
| fixed_rate      | 否    | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24，最小值5）。<br>最小值：1<br>最大值：60                                    |
| fixed_rate_unit | 否    | String  | 时间单位。<br>枚举值：<br><ul style="list-style-type: none"> <li>• minute</li> <li>• hour</li> </ul>                                                          |

表 6-592 SqlNotificationSaveRule

| 参数            | 是否必选 | 参数类型                    | 描述                                                                                                            |
|---------------|------|-------------------------|---------------------------------------------------------------------------------------------------------------|
| language      | 是    | String                  | 首选项对应的语言<br>最小长度：0<br>最大长度：10<br>枚举值： <ul style="list-style-type: none"><li>• zh-cn</li><li>• en-us</li></ul> |
| timezone      | 否    | String                  | 首选项对应的时区信息<br>最小长度：0<br>最大长度：1024                                                                             |
| user_name     | 是    | String                  | 用户名<br>最小长度：1<br>最大长度：1024                                                                                    |
| topics        | 是    | Array of Topics objects | 主题信息                                                                                                          |
| template_name | 是    | String                  | 消息模板名称                                                                                                        |

表 6-593 Topics

| 参数           | 是否必选 | 参数类型    | 描述                                                   |
|--------------|------|---------|------------------------------------------------------|
| name         | 是    | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                         |
| topic_urn    | 是    | String  | Topic 的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | 否    | String  | Topic 的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | 否    | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                        |

## 响应参数

状态码：200

表 6-594 响应 Body 参数

| 参数                     | 参数类型   | 描述                           |
|------------------------|--------|------------------------------|
| keywords_alarm_rule_id | String | 告警规则id<br>最小长度：36<br>最大长度：36 |

状态码：400

表 6-595 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

状态码：500

表 6-596 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

创建关键词告警规则

POST https://{endpoint}/v2/{project\_id}/lts/alarms/keywords-alarm-rule

```
{
 "keywords_alarm_rule_name": "huawei",
 "keywords_alarm_rule_description": "huawei",
 "keywords_requests": [{
 "log_stream_id": "1",
 "log_group_id": "1",
 "keywords": "huawei",
```

```
"condition": ">",
"number": "100",
"search_time_range": 10,
"search_time_range_unit": "minute"
}],
"frequency": {
 "type": "FIXED_RATE",
 "cron_expr": "",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 10,
 "fixed_rate_unit": "minute"
},
"keywords_alarm_level": "Critical",
"keywords_alarm_send": true,
"domain_id": "",
"notification_frequency": 5,
"alarm_action_rule_name": "",
"notification_save_rule": {
 "language": "zh-cn",
 "timezone": "Asia/Shanghai",
 "user_name": "huawei",
 "template_name": "消息模板名称",
 "topics": [{
 "name": "huawei",
 "topic_urn": "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name": "",
 "push_policy": 0
 }]
}
}
```

## 响应示例

### 状态码：200

请求响应成功。

```
{
 "keywords_alarm_rule_id": ""
}
```

### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code": "LTS.2005",
 "error_msg": "Alarm rule params validator error."
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2001",
 "error_msg": "Failed to create alarm rule."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

创建关键词告警规则

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateKeywordsAlarmRuleSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();

 CreateKeywordsAlarmRuleRequest request = new CreateKeywordsAlarmRuleRequest();
 CreateKeywordsAlarmRuleRequestBody body = new CreateKeywordsAlarmRuleRequestBody();
 List<Topics> listNotificationSaveRuleTopics = new ArrayList<>();
 listNotificationSaveRuleTopics.add(
 new Topics()
 .withName("huawei")
 .withTopicUrn("urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei")
 .withDisplayName("")
 .withPushPolicy(0)
);
 SqlNotificationSaveRule notificationSaveRulebody = new SqlNotificationSaveRule();
 notificationSaveRulebody.withLanguage(SqlNotificationSaveRule.LanguageEnum.fromValue("zh-cn"))
 .withTimezone("Asia/Shanghai")
 .withUserName("huawei")
 .withTopics(listNotificationSaveRuleTopics)
 .withTemplateName("消息模板名称");
 Frequency frequencybody = new Frequency();
 frequencybody.withType(Frequency.TypeEnum.fromValue("FIXED_RATE"))
 .withCronExpr("")
 .withHourOfDay(0)
 .withDayOfWeek(0)
 .withFixedRate(10)
 .withFixedRateUnit(Frequency.FixedRateUnitEnum.fromValue("minute"));
 List<KeywordsRequest> listbodyKeywordsRequests = new ArrayList<>();
 listbodyKeywordsRequests.add(
 new KeywordsRequest()
 .withLogStreamId("1")
 .withLogGroupId("1")
 .withKeywords("huawei")
 .withCondition(KeywordsRequest.ConditionEnum.fromValue(">"))
 .withNumber(100)
 .withSearchTimeRange(10)
 .withSearchTimeRangeUnit(KeywordsRequest.SearchTimeRangeUnitEnum.fromValue("minute"))
);
 }
};
```

```
body.withAlarmActionRuleName("");

body.withNotificationFrequency(CreateKeywordsAlarmRuleRequestBody.NotificationFrequencyEnum.NUMBER_5);
body.withNotificationSaveRule(notificationSaveRulebody);
body.withDomainId("");
body.withKeywordsAlarmSend(true);

body.withKeywordsAlarmLevel(CreateKeywordsAlarmRuleRequestBody.KeywordsAlarmLevelEnum.fromValue("Critical"));
body.withFrequency(frequencybody);
body.withKeywordsRequests(listbodyKeywordsRequests);
body.withKeywordsAlarmRuleDescription("huawei");
body.withKeywordsAlarmRuleName("huawei");
request.withBody(body);
try {
 CreateKeywordsAlarmRuleResponse response = client.createKeywordsAlarmRule(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 创建关键词告警规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateKeywordsAlarmRuleRequest()
 listTopicsNotificationSaveRule = [
 Topics(
 name="huawei",
 topic_urn="urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 display_name="",
 push_policy=0
)
]
 request.set_topics(listTopicsNotificationSaveRule)
 response = client.create_keywords_alarm_rule(request)
 print(response)
```

```
)
]
notificationSaveRulebody = SqlNotificationSaveRule(
 language="zh-cn",
 timezone="Asia/Shanghai",
 user_name="huawei",
 topics=listTopicsNotificationSaveRule,
 template_name="消息模板名称"
)
frequencybody = Frequency(
 type="FIXED_RATE",
 cron_expr="",
 hour_of_day=0,
 day_of_week=0,
 fixed_rate=10,
 fixed_rate_unit="minute"
)
listKeywordsRequestsbody = [
 KeywordsRequest(
 log_stream_id="1",
 log_group_id="1",
 keywords="huawei",
 condition=">",
 number=100,
 search_time_range=10,
 search_time_range_unit="minute"
)
]
request.body = CreateKeywordsAlarmRuleRequestBody(
 alarm_action_rule_name="",
 notification_frequency=5,
 notification_save_rule=notificationSaveRulebody,
 domain_id="",
 keywords_alarm_send=True,
 keywords_alarm_level="Critical",
 frequency=frequencybody,
 keywords_requests=listKeywordsRequestsbody,
 keywords_alarm_rule_description="huawei",
 keywords_alarm_rule_name="huawei"
)
response = client.create_keywords_alarm_rule(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建关键词告警规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
```

```
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.CreateKeywordsAlarmRuleRequest{
 displayNameTopics:= ""
 pushPolicyTopics:= int32(0)
 var listTopicsNotificationSaveRule = []model.Topics{
 {
 Name: "huawei",
 TopicUrn: "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 DisplayName: &displayNameTopics,
 PushPolicy: &pushPolicyTopics,
 },
 },
 timezoneNotificationSaveRule:= "Asia/Shanghai"
 notificationSaveRulebody := &model.SqlNotificationSaveRule{
 Language: model.GetSqlNotificationSaveRuleLanguageEnum().ZH_CN,
 Timezone: &timezoneNotificationSaveRule,
 UserName: "huawei",
 Topics: listTopicsNotificationSaveRule,
 TemplateName: "消息模板名称",
 }
 cronExprFrequency:= ""
 hourOfDayFrequency:= int32(0)
 dayOfWeekFrequency:= int32(0)
 fixedRateFrequency:= int32(10)
 fixedRateUnitFrequency:= model.GetFrequencyFixedRateUnitEnum().MINUTE
 frequencybody := &model.Frequency{
 Type: model.GetFrequencyTypeEnum().FIXED_RATE,
 CronExpr: &cronExprFrequency,
 HourOfDay: &hourOfDayFrequency,
 DayOfWeek: &dayOfWeekFrequency,
 FixedRate: &fixedRateFrequency,
 FixedRateUnit: &fixedRateUnitFrequency,
 }
 var listKeywordsRequestsbody = []model.KeywordsRequest{
 {
 LogStreamId: "1",
 LogGroupid: "1",
 Keywords: "huawei",
 Condition: model.GetKeywordsRequestConditionEnum().GREATER_THAN,
 Number: int32(100),
 SearchTimeRange: int32(10),
 SearchTimeRangeUnit: model.GetKeywordsRequestSearchTimeRangeUnitEnum().MINUTE,
 },
 },
 alarmActionRuleNameCreateKeywordsAlarmRuleRequestBody:= ""
 keywordsAlarmRuleDescriptionCreateKeywordsAlarmRuleRequestBody:= "huawei"
 request.Body = &model.CreateKeywordsAlarmRuleRequestBody{
 AlarmActionRuleName: &alarmActionRuleNameCreateKeywordsAlarmRuleRequestBody,
 NotificationFrequency:
model.GetCreateKeywordsAlarmRuleRequestBodyNotificationFrequencyEnum().E_5,
 NotificationSaveRule: notificationSaveRulebody,
 DomainId: "",
 KeywordsAlarmSend: true,
 KeywordsAlarmLevel:
model.GetCreateKeywordsAlarmRuleRequestBodyKeywordsAlarmLevelEnum().CRITICAL,
```

```
Frequency: frequencybody,
KeywordsRequests: listKeywordsRequestsbody,
KeywordsAlarmRuleDescription:
&keywordsAlarmRuleDescriptionCreateKeywordsAlarmRuleRequestBody,
KeywordsAlarmRuleName: "huawei",
}
response, err := client.CreateKeywordsAlarmRule(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

## 6.13.2 修改关键词告警规则

### 功能介绍

该接口用于修改关键词告警。

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v2/{project\_id}/lts/alarms/keywords-alarm-rule

表 6-597 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-598 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

表 6-599 请求 Body 参数

| 参数                       | 是否必选 | 参数类型   | 描述                                               |
|--------------------------|------|--------|--------------------------------------------------|
| keywords_alarm_rule_id   | 是    | String | 关键词告警规则id<br>最小长度: 36<br>最大长度: 36                |
| keywords_alarm_rule_name | 是    | String | 规则原始名称 ( 不支持修改首次创建的原始名称 )<br>最小长度: 1<br>最大长度: 64 |
| alarm_rule_alias         | 否    | String | 规则名称<br>最小长度: 1<br>最大长度: 64                      |

| 参数                              | 是否必选 | 参数类型                                    | 描述                                                                                                                                                 |
|---------------------------------|------|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords_alarm_rule_description | 否    | String                                  | 关键词告警信息描述<br>最小长度：0<br>最大长度：64                                                                                                                     |
| keywords_requests               | 是    | Array of <b>KeywordsRequest</b> objects | 关键词详细信息                                                                                                                                            |
| frequency                       | 是    | <b>Frequency</b> object                 | 告警统计周期                                                                                                                                             |
| keywords_alarm_level            | 是    | String                                  | 告警级别<br>枚举值：<br><ul style="list-style-type: none"> <li>• Info</li> <li>• Minor</li> <li>• Major</li> <li>• Critical</li> </ul>                     |
| keywords_alarm_send             | 是    | Boolean                                 | 是否发送                                                                                                                                               |
| keywords_alarm_send_code        | 是    | Integer                                 | 发送主题 0:不变 1:新增 2:修改 3:删除<br>最小值：0<br>最大值：3<br>枚举值：<br><ul style="list-style-type: none"> <li>• 0</li> <li>• 1</li> <li>• 2</li> <li>• 3</li> </ul> |
| domain_id                       | 是    | String                                  | domainId<br>最小长度：32<br>最大长度：32                                                                                                                     |
| notification_save_rule          | 否    | <b>SqlNotificationSaveRule</b> object   | 通知主题                                                                                                                                               |
| trigger_condition_count         | 否    | Integer                                 | 触发条件：触发次数；默认为1                                                                                                                                     |
| trigger_condition_frequency     | 否    | Integer                                 | 触发条件：触发周期；默认为1                                                                                                                                     |
| whether_recover_policy          | 否    | Boolean                                 | 是否打开恢复通知；默认false                                                                                                                                   |

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                                                                           |
|------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| recovery_policy        | 否    | Integer | 恢复策略周期；默认为3                                                                                                                                                                  |
| notification_frequency | 是    | Integer | 通知频率,单位(分钟)<br>枚举值: <ul style="list-style-type: none"> <li>• 0</li> <li>• 5</li> <li>• 10</li> <li>• 15</li> <li>• 30</li> <li>• 60</li> <li>• 180</li> <li>• 360</li> </ul> |
| alarm_action_rule_name | 否    | String  | 告警行动规则名称<br><b>说明</b><br>alarm_action_rule_name和notification_save_rule可以选填一个，如果都填，优先选择alarm_action_rule_name                                                                 |

表 6-600 KeywordsRequest

| 参数              | 是否必选 | 参数类型   | 描述                                          |
|-----------------|------|--------|---------------------------------------------|
| log_stream_id   | 是    | String | 日志流id<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| log_stream_name | 否    | String | 日志流名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>  |
| log_group_id    | 是    | String | 日志组id<br>最小长度: <b>36</b><br>最大长度: <b>36</b> |
| log_group_name  | 否    | String | 日志组名称<br>最小长度: <b>1</b><br>最大长度: <b>64</b>  |
| keywords        | 是    | String | 关键词<br>最小长度: <b>1</b><br>最大长度: <b>1024</b>  |

| 参数                     | 是否必选 | 参数类型    | 描述                                                                                                                       |
|------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------|
| condition              | 是    | String  | 条件<br>枚举值：<br><ul style="list-style-type: none"> <li>• &gt;=</li> <li>• &lt;=</li> <li>• &lt;</li> <li>• &gt;</li> </ul> |
| number                 | 是    | Integer | 行数<br>最小值：1<br>最大值：2147483647                                                                                            |
| search_time_range      | 是    | Integer | 查询执行任务时最近数据的时间范围，最大值为60 minute：最小值：1；最大值为60 hour：最小值：1；最大值为24                                                            |
| search_time_range_unit | 是    | String  | 查询时间单位<br>枚举值：<br><ul style="list-style-type: none"> <li>• minute</li> </ul>                                             |

表 6-601 Frequency

| 参数          | 是否必选 | 参数类型    | 描述                                                                                                                                                   |
|-------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| type        | 是    | String  | 时间类型。<br>枚举值：<br><ul style="list-style-type: none"> <li>• CRON</li> <li>• HOURLY</li> <li>• DAILY</li> <li>• WEEKLY</li> <li>• FIXED_RATE</li> </ul> |
| cron_expr   | 否    | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                          |
| hour_of_day | 否    | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。<br>DAILY：最小值：0，最大值：23<br>WEEKLY：最小值：0，最大值：23                                                                         |

| 参数              | 是否必选 | 参数类型    | 描述                                                                                                                |
|-----------------|------|---------|-------------------------------------------------------------------------------------------------------------------|
| day_of_week     | 否    | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。<br>最小值：1<br>最大值：7                                                                   |
| fixed_rate      | 否    | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24，最小值5）。<br>最小值：1<br>最大值：60 |
| fixed_rate_unit | 否    | String  | 时间单位。<br>枚举值：<br><ul style="list-style-type: none"> <li>minute</li> <li>hour</li> </ul>                           |

表 6-602 SqlNotificationSaveRule

| 参数            | 是否必选 | 参数类型                    | 描述                                                                                                              |
|---------------|------|-------------------------|-----------------------------------------------------------------------------------------------------------------|
| language      | 是    | String                  | 首选项对应的语言<br>最小长度：0<br>最大长度：10<br>枚举值：<br><ul style="list-style-type: none"> <li>zh-cn</li> <li>en-us</li> </ul> |
| timezone      | 否    | String                  | 首选项对应的时区信息<br>最小长度：0<br>最大长度：1024                                                                               |
| user_name     | 是    | String                  | 用户名<br>最小长度：1<br>最大长度：1024                                                                                      |
| topics        | 是    | Array of Topics objects | 主题信息                                                                                                            |
| template_name | 是    | String                  | 消息模板名称                                                                                                          |

表 6-603 Topics

| 参数           | 是否必选 | 参数类型    | 描述                                                  |
|--------------|------|---------|-----------------------------------------------------|
| name         | 是    | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                        |
| topic_urn    | 是    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | 否    | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | 否    | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

## 响应参数

状态码：200

表 6-604 响应 Body 参数

| 参数                              | 参数类型                                             | 描述                             |
|---------------------------------|--------------------------------------------------|--------------------------------|
| keywords_alarm_rule_id          | String                                           | 关键词告警id<br>最小长度：36<br>最大长度：36  |
| keywords_alarm_rule_name        | String                                           | 原始规则名称<br>最小长度：1<br>最大长度：64    |
| alarm_rule_alias                | String                                           | 规则名称                           |
| keywords_alarm_rule_description | String                                           | 关键词告警信息描述<br>最小长度：0<br>最大长度：64 |
| keywords_requests               | Array of <a href="#">KeywordsResBody</a> objects | 关键词详细信息                        |
| frequency                       | <a href="#">FrequencyResBody</a> object          | 告警统计周期                         |

| 参数                   | 参数类型                           | 描述                                                                                                                                                 |
|----------------------|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| keywords_alarm_level | String                         | 告警级别<br>枚举值: <ul style="list-style-type: none"><li>• <b>Info</b></li><li>• <b>Minor</b></li><li>• <b>Major</b></li><li>• <b>Critical</b></li></ul> |
| keywords_alarm_send  | Boolean                        | 是否发送                                                                                                                                               |
| domain_id            | String                         | domainId<br>最小长度: <b>32</b><br>最大长度: <b>32</b>                                                                                                     |
| create_time          | Long                           | 创建时间(毫秒时间戳)<br>最小值: <b>13</b><br>最大值: <b>13</b>                                                                                                    |
| update_time          | Long                           | 更新时间(毫秒时间戳)<br>最小值: <b>13</b><br>最大值: <b>13</b>                                                                                                    |
| language             | String                         | 邮件附加信息语言<br>最小长度: <b>1</b><br>最大长度: <b>1024</b><br>枚举值: <ul style="list-style-type: none"><li>• <b>zh-cn</b></li><li>• <b>en-us</b></li></ul>      |
| projectId            | String                         | 项目id                                                                                                                                               |
| topics               | Array of <b>Topics</b> objects | 通知主题                                                                                                                                               |
| condition_expression | String                         | 情况表述                                                                                                                                               |
| indexId              | String                         | 索引id                                                                                                                                               |

| 参数                     | 参数类型    | 描述                                                                                                                                                                  |
|------------------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| notification_frequency | Integer | 通知频率,单位(分钟)<br>枚举值: <ul style="list-style-type: none"><li>• 0</li><li>• 5</li><li>• 10</li><li>• 15</li><li>• 30</li><li>• 60</li><li>• 180</li><li>• 360</li></ul> |
| alarm_action_rule_name | String  | 告警行动规则名称<br>说明<br>alarm_action_rule_name和notification_save_rule可以选填一个, 如果都填, 优先选择alarm_action_rule_name                                                             |

表 6-605 KeywordsResBody

| 参数              | 参数类型   | 描述                            |
|-----------------|--------|-------------------------------|
| log_stream_id   | String | 日志流id<br>最小长度: 36<br>最大长度: 36 |
| log_stream_name | String | 日志流名称<br>最小长度: 1<br>最大长度: 64  |
| log_group_id    | String | 日志组id<br>最小长度: 36<br>最大长度: 36 |
| log_group_name  | String | 日志组名称<br>最小长度: 1<br>最大长度: 64  |
| keywords        | String | 关键词<br>最小长度: 1<br>最大长度: 1024  |

| 参数                     | 参数类型    | 描述                                                                                                               |
|------------------------|---------|------------------------------------------------------------------------------------------------------------------|
| condition              | String  | 条件<br>枚举值： <ul style="list-style-type: none"><li>• &gt;=</li><li>• &lt;=</li><li>• &lt;</li><li>• &gt;</li></ul> |
| number                 | Integer | 行数<br>最小长度：1<br>最大长度：2147483647                                                                                  |
| search_time_range      | Integer | 查询执行任务时最近数据的时间范围，最大值为60<br>最小长度：1<br>最大长度：60                                                                     |
| search_time_range_unit | String  | 查询时间单位                                                                                                           |

表 6-606 FrequencyRespBody

| 参数          | 参数类型    | 描述                                                                                                                                          |
|-------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------|
| type        | String  | 时间类型。<br>枚举值： <ul style="list-style-type: none"><li>• CRON</li><li>• HOURLY</li><li>• DAILY</li><li>• WEEKLY</li><li>• FIXED_RATE</li></ul> |
| cron_expr   | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                 |
| hour_of_day | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。                                                                                                             |
| day_of_week | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。                                                                                                               |
| fixed_rate  | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24）。                                                   |

| 参数              | 参数类型   | 描述                                                                                          |
|-----------------|--------|---------------------------------------------------------------------------------------------|
| fixed_rate_unit | String | 时间单位枚举值：<br>枚举值： <ul style="list-style-type: none"> <li>• minute</li> <li>• hour</li> </ul> |

表 6-607 Topics

| 参数           | 参数类型    | 描述                                                   |
|--------------|---------|------------------------------------------------------|
| name         | String  | 主题名称。<br>最小长度：1<br>最大长度：1024                         |
| topic_urn    | String  | Topic 的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | String  | Topic 的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                        |

状态码：400

表 6-608 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                        |
|------------|--------|-------------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                      |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"> <li>• Invalid projectId</li> </ul> |

状态码：500

表 6-609 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

### 修改关键词告警规则

PUT https://{endpoint}/v2/{project\_id}/lts/alarms/keywords-alarm-rule

```
{
 "keywords_alarm_rule_id": "",
 "keywords_alarm_rule_name": "huawei",
 "alarm_rule_alias": "zhangsan",
 "keywords_alarm_rule_description": "huawei",
 "keywords_requests": [{
 "log_stream_id": "1",
 "log_group_id": "1",
 "keywords": "huawei",
 "condition": ">",
 "number": "100",
 "search_time_range": 10,
 "search_time_range_unit": "minute"
 }],
 "frequency": {
 "type": "FIXED_RATE",
 "cron_expr": "",
 "hour_of_day": 0,
 "day_of_week": 0,
 "fixed_rate": 10,
 "fixed_rate_unit": "minute"
 },
 "keywords_alarm_level": "Critical",
 "keywords_alarm_send": true,
 "keywords_alarm_send_code": "2",
 "domain_id": "",
 "notification_frequency": 5,
 "alarm_action_rule_name": "",
 "notification_save_rule": {
 "language": "zh-cn",
 "timezone": "Asia/Shanghai",
 "user_name": "huawei",
 "template_name": "消息模板名称",
 "topics": [{
 "name": "huawei",
 "topic_urn": "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name": "",
 "push_policy": 0
 }]
 }
}
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "keywords_alarm_rule_id" : "",
 "keywords_alarm_rule_name" : "huawei",
 "keywords_alarm_rule_description" : "huawei",
 "alarm_rule_alias" : "zhangsan",
 "keywords_requests" : [{
 "log_stream_id" : "1",
 "log_stream_name" : "huawei",
 "log_group_name" : "huawei",
 "log_group_id" : "1",
 "keywords" : "huawei",
 "condition" : ">",
 "number" : "100",
 "search_time_range" : 10,
 "search_time_range_unit" : "minute"
 }],
 "frequency" : {
 "type" : "FIXED_RATE",
 "cron_expr" : "",
 "hour_of_day" : 0,
 "day_of_week" : 0,
 "fixed_rate" : 10,
 "fixed_rate_unit" : "minute"
 },
 "keywords_alarm_level" : "Critical",
 "keywords_alarm_send" : false,
 "domain_id" : "",
 "notification_frequency" : 5,
 "alarm_action_rule_name" : "",
 "topics" : [{
 "name" : "huawei",
 "topic_urn" : "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 "display_name" : "",
 "push_policy" : 0
 }],
 "language" : "zh-cn"
}
```

#### 状态码：400

BadRequest 非法请求 建议根据error\_msg直接修改该请求。

```
{
 "error_code" : "LTS.2005",
 "error_msg" : "Alarm rule params validator error."
}
```

#### 状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code" : "LTS.2003",
 "error_msg" : "Failed to update alarm rule."
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

修改关键词告警规则

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
```

```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateKeywordsAlarmRuleSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateKeywordsAlarmRuleRequest request = new UpdateKeywordsAlarmRuleRequest();
 UpdateKeywordsAlarmRuleRequestBody body = new UpdateKeywordsAlarmRuleRequestBody();
 List<Topics> listNotificationSaveRuleTopics = new ArrayList<>();
 listNotificationSaveRuleTopics.add(
 new Topics()
 .withName("huawei")
 .withTopicUrn("urn:smn:cn:north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei")
 .withDisplayName("")
 .withPushPolicy(0)
);
 SqlNotificationSaveRule notificationSaveRulebody = new SqlNotificationSaveRule();
 notificationSaveRulebody.withLanguage(SqlNotificationSaveRule.LanguageEnum.fromValue("zh-cn"))
 .withTimezone("Asia/Shanghai")
 .withUserName("huawei")
 .withTopics(listNotificationSaveRuleTopics)
 .withTemplateName("消息模板名称");
 Frequency frequencybody = new Frequency();
 frequencybody.withType(Frequency.TypeEnum.fromValue("FIXED_RATE"))
 .withCronExpr("")
 .withHourOfDay(0)
 .withDayOfWeek(0)
 .withFixedRate(10)
 .withFixedRateUnit(Frequency.FixedRateUnitEnum.fromValue("minute"));
 List<KeywordsRequest> listbodyKeywordsRequests = new ArrayList<>();
 listbodyKeywordsRequests.add(
 new KeywordsRequest()
 .withLogStreamId("1")
 .withLogGroupId("1")
 .withKeywords("huawei")
 .withCondition(KeywordsRequest.ConditionEnum.fromValue(">"))
 .withNumber(100)
 .withSearchTimeRange(10)
 .withSearchTimeRangeUnit(KeywordsRequest.SearchTimeRangeUnitEnum.fromValue("minute"))
);
 body.withAlarmActionRuleName("");

 body.withNotificationFrequency(UpdateKeywordsAlarmRuleRequestBody.NotificationFrequencyEnum.NUMB
```

```
ER_5);
 body.withNotificationSaveRule(notificationSaveRulebody);
 body.withDomainId("");

body.withKeywordsAlarmSendCode(UpdateKeywordsAlarmRuleRequestBody.KeywordsAlarmSendCodeEnum.
NUMBER_2);
 body.withKeywordsAlarmSend(true);

body.withKeywordsAlarmLevel(UpdateKeywordsAlarmRuleRequestBody.KeywordsAlarmLevelEnum.fromValu
e("Critical"));
 body.withFrequency(frequencybody);
 body.withKeywordsRequests(listbodyKeywordsRequests);
 body.withKeywordsAlarmRuleDescription("huawei");
 body.withAlarmRuleAlias("zhangsang");
 body.withKeywordsAlarmRuleName("huawei");
 body.withKeywordsAlarmRuleId("");
 request.withBody(body);
 try {
 UpdateKeywordsAlarmRuleResponse response = client.updateKeywordsAlarmRule(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 修改关键词告警规则

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateKeywordsAlarmRuleRequest()
 listTopicsNotificationSaveRule = [
 Topics(
 name="huawei",
 topic_urn="urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
```

```
 display_name="",
 push_policy=0
)
]
notificationSaveRulebody = SqlNotificationSaveRule(
 language="zh-cn",
 timezone="Asia/Shanghai",
 user_name="huawei",
 topics=listTopicsNotificationSaveRule,
 template_name="消息模板名称"
)
frequencybody = Frequency(
 type="FIXED_RATE",
 cron_expr="",
 hour_of_day=0,
 day_of_week=0,
 fixed_rate=10,
 fixed_rate_unit="minute"
)
listKeywordsRequestsbody = [
 KeywordsRequest(
 log_stream_id="1",
 log_group_id="1",
 keywords="huawei",
 condition=">",
 number=100,
 search_time_range=10,
 search_time_range_unit="minute"
)
]
request.body = UpdateKeywordsAlarmRuleRequestBody(
 alarm_action_rule_name="",
 notification_frequency=5,
 notification_save_rule=notificationSaveRulebody,
 domain_id="",
 keywords_alarm_send_code=2,
 keywords_alarm_send=True,
 keywords_alarm_level="Critical",
 frequency=frequencybody,
 keywords_requests=listKeywordsRequestsbody,
 keywords_alarm_rule_description="huawei",
 alarm_rule_alias="zhagsan",
 keywords_alarm_rule_name="huawei",
 keywords_alarm_rule_id=""
)
response = client.update_keywords_alarm_rule(request)
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 修改关键词告警规则

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD\_SDK\_AK and CLOUD\_SDK\_SK in the local environment

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.UpdateKeywordsAlarmRuleRequest{
 displayNameTopics:= ""
 pushPolicyTopics:= int32(0)
 var listTopicsNotificationSaveRule = []model.Topics{
 {
 Name: "huawei",
 TopicUrn: "urn:smn:cn-north-7:1b06fc5dc0814a4da1594a9ade9cb93c:huawei",
 DisplayName: &displayNameTopics,
 PushPolicy: &pushPolicyTopics,
 },
 },
 timezoneNotificationSaveRule:= "Asia/Shanghai"
 notificationSaveRulebody := &model.SqlNotificationSaveRule{
 Language: model.GetSqlNotificationSaveRuleLanguageEnum().ZH_CN,
 Timezone: &timezoneNotificationSaveRule,
 UserName: "huawei",
 Topics: listTopicsNotificationSaveRule,
 TemplateName: "消息模板名称",
 }
 cronExprFrequency:= ""
 hourOfDayFrequency:= int32(0)
 dayOfWeekFrequency:= int32(0)
 fixedRateFrequency:= int32(10)
 fixedRateUnitFrequency:= model.GetFrequencyFixedRateUnitEnum().MINUTE
 frequencybody := &model.Frequency{
 Type: model.GetFrequencyTypeEnum().FIXED_RATE,
 CronExpr: &cronExprFrequency,
 HourOfDay: &hourOfDayFrequency,
 DayOfWeek: &dayOfWeekFrequency,
 FixedRate: &fixedRateFrequency,
 FixedRateUnit: &fixedRateUnitFrequency,
 }
 var listKeywordsRequestsbody = []model.KeywordsRequest{
 {
 LogStreamId: "1",
 LogGroupId: "1",
 Keywords: "huawei",
 Condition: model.GetKeywordsRequestConditionEnum().GREATER_THAN,
 Number: int32(100),
 SearchTimeRange: int32(10),
 SearchTimeRangeUnit: model.GetKeywordsRequestSearchTimeRangeUnitEnum().MINUTE,
 },
 },
 alarmActionRuleNameUpdateKeywordsAlarmRuleRequestBody:= ""
 keywordsAlarmRuleDescriptionUpdateKeywordsAlarmRuleRequestBody:= "huawei"
 alarmRuleAliasUpdateKeywordsAlarmRuleRequestBody:= "zhangsan"
 request.Body = &model.UpdateKeywordsAlarmRuleRequestBody{
 AlarmActionRuleName: &alarmActionRuleNameUpdateKeywordsAlarmRuleRequestBody,
 NotificationFrequency:
```

```
model.GetUpdateKeywordsAlarmRuleRequestBodyNotificationFrequencyEnum().E_5,
 NotificationSaveRule: notificationSaveRulebody,
 DomainId: "",
 KeywordsAlarmSendCode:
model.GetUpdateKeywordsAlarmRuleRequestBodyKeywordsAlarmSendCodeEnum().E_2,
 KeywordsAlarmSend: true,
 KeywordsAlarmLevel:
model.GetUpdateKeywordsAlarmRuleRequestBodyKeywordsAlarmLevelEnum().CRITICAL,
 Frequency: frequencybody,
 KeywordsRequests: listKeywordsRequestsbody,
 KeywordsAlarmRuleDescription:
&keywordsAlarmRuleDescriptionUpdateKeywordsAlarmRuleRequestBody,
 AlarmRuleAlias: &alarmRuleAliasUpdateKeywordsAlarmRuleRequestBody,
 KeywordsAlarmRuleName: "huawei",
 KeywordsAlarmRuleId: "",
}
response, err := client.UpdateKeywordsAlarmRule(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                    |
|-----|---------------------------------------|
| 200 | 请求响应成功。                               |
| 400 | BadRequest 非法请求 建议根据error_msg直接修改该请求。 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。                |

## 错误码

请参见[错误码](#)。

### 6.13.3 查询关键词告警规则

#### 功能介绍

该接口用于查询关键词告警。

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v2/{project\_id}/lts/alarms/keywords-alarm-rule

表 6-610 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-611 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

## 响应参数

状态码: 200

表 6-612 响应 Body 参数

| 参数                   | 参数类型                                                       | 描述   |
|----------------------|------------------------------------------------------------|------|
| keywords_alarm_rules | Array of <a href="#">KeywordsAlarmRuleRespList</a> objects | 项目id |

表 6-613 KeywordsAlarmRuleRespList

| 参数                              | 参数类型                                             | 描述                                                                                                                     |
|---------------------------------|--------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| projectId                       | String                                           | 项目id<br>最小长度: 32<br>最大长度: 32                                                                                           |
| keywords_alarm_rule_id          | String                                           | 关键词告警id<br>最小长度: 36<br>最大长度: 36                                                                                        |
| keywords_alarm_rule_name        | String                                           | 关键词告警名称<br>最小长度: 1<br>最大长度: 64                                                                                         |
| keywords_alarm_rule_description | String                                           | 关键词告警信息描述<br>最小长度: 0<br>最大长度: 64                                                                                       |
| condition_expression            | String                                           | 条件<br>最小长度: 0<br>最大长度: 64                                                                                              |
| keywords_requests               | Array of <a href="#">KeywordsRequest</a> objects | 关键词详细信息                                                                                                                |
| frequency                       | <a href="#">Frequency</a> object                 | 告警统计周期                                                                                                                 |
| keywords_alarm_level            | String                                           | 告警级别<br>枚举值: <ul style="list-style-type: none"><li>• Info</li><li>• Minor</li><li>• Major</li><li>• Critical</li></ul> |
| keywords_alarm_send             | Boolean                                          | 是否发送                                                                                                                   |
| domain_id                       | String                                           | domainId<br>最小长度: 32<br>最大长度: 32                                                                                       |
| create_time                     | Long                                             | 创建时间 (毫秒时间戳)<br>最小值: 13<br>最大值: 13                                                                                     |

| 参数                          | 参数类型                           | 描述                                                                                                                                                                              |
|-----------------------------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| update_time                 | Long                           | 更新时间（毫秒时间戳）<br>最小值：13<br>最大值：13                                                                                                                                                 |
| topics                      | Array of <b>Topics</b> objects | 通知主题                                                                                                                                                                            |
| template_name               | String                         | 消息模板名称                                                                                                                                                                          |
| status                      | String                         | 告警状态<br>枚举值：<br><ul style="list-style-type: none"> <li>● <b>RUNNING</b> 启用</li> <li>● <b>STOPPING</b> 停止</li> </ul>                                                             |
| trigger_condition_count     | Integer                        | 触发条件：触发周期；默认为1                                                                                                                                                                  |
| trigger_condition_frequency | Integer                        | 触发条件：触发次数；默认为1                                                                                                                                                                  |
| whether_recovery_policy     | Boolean                        | 是否打开恢复通知；默认false                                                                                                                                                                |
| recovery_policy             | Integer                        | 恢复策略周期；默认为3                                                                                                                                                                     |
| notification_frequency      | Integer                        | 通知频率,单位(分钟)<br>枚举值：<br><ul style="list-style-type: none"> <li>● 0</li> <li>● 5</li> <li>● 10</li> <li>● 15</li> <li>● 30</li> <li>● 60</li> <li>● 180</li> <li>● 360</li> </ul> |
| alarm_action_rule_name      | String                         | 告警行动规则名称<br><b>说明</b><br>alarm_action_rule_name和notification_save_rule可以选填一个，如果都填，优先选择alarm_action_rule_name                                                                    |

表 6-614 KeywordsRequest

| 参数                     | 参数类型    | 描述                                                                                                               |
|------------------------|---------|------------------------------------------------------------------------------------------------------------------|
| log_stream_id          | String  | 日志流id<br>最小长度: 36<br>最大长度: 36                                                                                    |
| log_stream_name        | String  | 日志流名称<br>最小长度: 1<br>最大长度: 64                                                                                     |
| log_group_id           | String  | 日志组id<br>最小长度: 36<br>最大长度: 36                                                                                    |
| log_group_name         | String  | 日志组名称<br>最小长度: 1<br>最大长度: 64                                                                                     |
| keywords               | String  | 关键词<br>最小长度: 1<br>最大长度: 1024                                                                                     |
| condition              | String  | 条件<br>枚举值: <ul style="list-style-type: none"><li>• &gt;=</li><li>• &lt;=</li><li>• &lt;</li><li>• &gt;</li></ul> |
| number                 | Integer | 行数<br>最小值: 1<br>最大值: 2147483647                                                                                  |
| search_time_range      | Integer | 查询执行任务时最近数据的时间范围,<br>最大值为60 minute: 最小值: 1; 最大值:<br>60 hour: 最小值: 1; 最大值: 24                                     |
| search_time_range_unit | String  | 查询时间单位<br>枚举值: <ul style="list-style-type: none"><li>• minute</li></ul>                                          |

表 6-615 Frequency

| 参数              | 参数类型    | 描述                                                                                                                                          |
|-----------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------|
| type            | String  | 时间类型。<br>枚举值： <ul style="list-style-type: none"><li>• CRON</li><li>• HOURLY</li><li>• DAILY</li><li>• WEEKLY</li><li>• FIXED_RATE</li></ul> |
| cron_expr       | String  | 当字段type为"CRON"时取该字段。<br>最小长度：1<br>最大长度：1024                                                                                                 |
| hour_of_day     | Integer | 当字段type为"DAILY"或者"WEEKLY"时取该字段。<br>DAILY：最小值：0，最大值：23<br>WEEKLY：最小值：0，最大值：23                                                                |
| day_of_week     | Integer | 当字段type为"WEEKLY"时取该字段（周日~周六）。<br>最小值：1<br>最大值：7                                                                                             |
| fixed_rate      | Integer | 当字段type为"FIXED_RATE"时取该字段（当fixed_rate_unit单位为minute，最大值60；当fixed_rate_unit单位为hour，最大值24，最小值5）。<br>最小值：1<br>最大值：60                           |
| fixed_rate_unit | String  | 时间单位。<br>枚举值： <ul style="list-style-type: none"><li>• minute</li><li>• hour</li></ul>                                                       |

表 6-616 Topics

| 参数   | 参数类型   | 描述                           |
|------|--------|------------------------------|
| name | String | 主题名称。<br>最小长度：1<br>最大长度：1024 |

| 参数           | 参数类型    | 描述                                                  |
|--------------|---------|-----------------------------------------------------|
| topic_urn    | String  | Topic的唯一的资源标识。<br>最小长度：1<br>最大长度：1024               |
| display_name | String  | Topic的显示名，推送邮件消息时，作为邮件发件人显示。<br>最小长度：1<br>最大长度：1024 |
| push_policy  | Integer | 消息推送的策略。<br>最小值：1<br>最大值：1024                       |

状态码：500

表 6-617 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

查询关键词告警规则

```
GET https://{endpoint}/v2/{project_id}/lts/alarms/keywords-alarm-rule
/v2/{project_id}/lts/alarms/keywords-alarm-rule
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "keywords_alarm_rules" : [{
 "projectId" : "string",
 "keywords_alarm_rule_id" : "string",
 "keywords_alarm_rule_name" : "string",
 "keywords_alarm_rule_description" : "string",
 "condition_expression" : "string",
 "keywords_requests" : [{
 "log_stream_id" : "string",
 "log_stream_name" : "string",
 "log_group_id" : "string",
```

```
"log_group_name" : "string",
"keywords" : "string",
"condition" : ">=",
"number" : 1,
"search_time_range" : 0,
"search_time_range_unit" : "minute"
}],
"frequency" : {
"type" : "CRON",
"cron_expr" : "string",
"hour_of_day" : 0,
"day_of_week" : 0,
"fixed_rate" : 0,
"fixed_rate_unit" : "minute"
},
"keywords_alarm_level" : "Info",
"keywords_alarm_send" : true,
"domain_id" : "string",
"create_time" : 0,
"update_time" : 0,
"template_name" : "消息模板名称",
"status" : "RUNNING",
"trigger_condition_count" : "1",
"trigger_condition_frequency" : "1",
"whether_recovery_policy" : false,
"recovery_policy" : "3",
"notification_frequency" : 5,
"alarm_action_rule_name" : "",
"topics" : [{
"name" : "string",
"topic_urn" : "string",
"display_name" : "string",
"push_policy" : 0
}]
}]
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错。

```
{
"error_code" : "LTS.2008",
"error_msg" : "Find Alarm rule failed."
}
```

**状态码**

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

**错误码**

请参见[错误码](#)。

## 6.13.4 删除关键词告警规则

### 功能介绍

该接口用于删除关键词告警。

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v2/{project\_id}/lts/alarms/keywords-alarm-rule/{keywords\_alarm\_rule\_id}

表 6-618 路径参数

| 参数                     | 是否必选 | 参数类型   | 描述                                                                            |
|------------------------|------|--------|-------------------------------------------------------------------------------|
| project_id             | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| keywords_alarm_rule_id | 是    | String | 关键词告警规则id。<br>最小长度：36<br>最大长度：36                                              |

### 请求参数

表 6-619 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：500

表 6-620 响应 Body 参数

| 参数         | 参数类型   | 描述                                                                                    |
|------------|--------|---------------------------------------------------------------------------------------|
| error_code | String | 错误码。                                                                                  |
| error_msg  | String | 调用失败响应信息描述。<br>枚举值： <ul style="list-style-type: none"><li>Invalid projectId</li></ul> |

## 请求示例

根据告警ID删除关键词告警规则

```
POST https://{endpoint}/v2/{project_id}/lts/alarms/keywords-alarm-rule/{keywords_alarm_rule_id}
/v2/{project_id}/lts/alarms/keywords-alarm-rule/{keywords_alarm_rule_id}
```

## 响应示例

状态码：500

表明服务端能被请求访问到，但是服务内部出错。

```
{
 "error_code": "LTS.2002",
 "error_msg": "Failed to update alarm rule."
}
```

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.14 告警列表

## 6.14.1 查询活动或历史告警列表

### 功能介绍

该接口用于查询告警列表

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/{domain\_id}/lts/alarms/sql-alarm/query

表 6-621 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| domain_id  | 是    | String | 账号ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-622 Query 参数

| 参数     | 是否必选 | 参数类型   | 描述                                                                                                          |
|--------|------|--------|-------------------------------------------------------------------------------------------------------------|
| type   | 是    | String | 是活动告警还是历史告警。<br>枚举值： <ul style="list-style-type: none"><li>• active_alert</li><li>• history_alert</li></ul> |
| marker | 否    | String | 取值为上一页数据的最后一条记录的id（填写上一页数据返回的previous_marker或者next_marker值）。<br>最小长度：0<br>最大长度：1000                         |

| 参数    | 是否必选 | 参数类型    | 描述                         |
|-------|------|---------|----------------------------|
| limit | 否    | Integer | 每页数据量<br>最小值：0<br>最大值：1000 |

## 请求参数

表 6-623 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                |
|--------------|------|--------|-----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，<br>获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/<br>json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-624 请求 Body 参数

| 参数                   | 是否必选 | 参数类型    | 描述                                      |
|----------------------|------|---------|-----------------------------------------|
| whether_custom_field | 是    | Boolean | 是否自定义查询时间段                              |
| start_time           | 否    | Long    | 自定义时间段开始时间(时间戳)<br>最小值：13<br>最大值：13     |
| end_time             | 否    | Long    | 自定义时间段结束时间(时间戳)<br>最小值：13<br>最大值：13     |
| time_range           | 否    | String  | 非自定义时间段时间范围(单位为分钟)<br>最小长度：1<br>最大长度：32 |
| search               | 否    | String  | 关键字检索条件<br>最小长度：1<br>最大长度：1024          |

| 参数              | 是否必选 | 参数类型               | 描述                                                                                                       |
|-----------------|------|--------------------|----------------------------------------------------------------------------------------------------------|
| alarm_level_ids | 否    | Array of strings   | 告警级别 ("Critical","Major","Minor","Info")<br><b>说明</b><br>alarmLevelIds为旧版参数，目前兼容该参数，建议使用 alarm_level_ids |
| sort            | 否    | <b>Sort</b> object | 排序检索条件                                                                                                   |
| step            | 否    | Integer            | 关键字检索条件                                                                                                  |

表 6-625 Sort

| 参数       | 是否必选 | 参数类型             | 描述                                                                                      |
|----------|------|------------------|-----------------------------------------------------------------------------------------|
| order_by | 是    | Array of strings | 排序字段                                                                                    |
| order    | 是    | String           | 排序顺序<br>枚举值：<br><ul style="list-style-type: none"> <li>• desc</li> <li>• asc</li> </ul> |

## 响应参数

状态码：200

表 6-626 响应 Body 参数

| 参数        | 参数类型                           | 描述                          |
|-----------|--------------------------------|-----------------------------|
| events    | Array of <b>Events</b> objects | 告警信息<br>最小长度：0<br>最大长度：2048 |
| page_info | <b>PageInfo</b> object         | 分页详情<br>最小长度：0<br>最大长度：2048 |

表 6-627 Events

| 参数          | 参数类型                         | 描述                                      |
|-------------|------------------------------|-----------------------------------------|
| annotations | <b>Annotations</b><br>object | 告警详情<br>最小长度: 0<br>最大长度: 5000           |
| metadata    | <b>Metadata</b> object       | 告警信息<br>最小长度: 0<br>最大长度: 2048           |
| arrives_at  | Long                         | 到达时间 (时间戳)<br>最小值: 13<br>最大值: 13        |
| ends_at     | Long                         | 告警清除时间 (时间戳)<br>最小值: 13<br>最大值: 13      |
| id          | String                       | 告警id<br>最小长度: 0<br>最大长度: 64             |
| starts_at   | Long                         | 告警产生时间 (时间戳)<br>最小值: 13<br>最大值: 13      |
| timeout     | Long                         | 告警自动清除时间 (时间戳)<br>最小值: 1<br>最大值: 13     |
| type        | String                       | 告警规则类型 (SQL/关键词)<br>最小长度: 0<br>最大长度: 16 |

表 6-628 Annotations

| 参数       | 参数类型   | 描述                                 |
|----------|--------|------------------------------------|
| message  | String | 告警列表详情<br>最小长度: 1<br>最大长度: 1024    |
| log_info | String | 日志组/流id名称<br>最小长度: 1<br>最大长度: 1024 |

| 参数              | 参数类型   | 描述                                        |
|-----------------|--------|-------------------------------------------|
| current_value   | String | 当前值<br>最小长度：1<br>最大长度：1024                |
| old_annotations | String | (sql/关键词)告警详情原始数据<br>最小长度：0<br>最大长度：10000 |

表 6-629 Metadata

| 参数                | 参数类型   | 描述                                     |
|-------------------|--------|----------------------------------------|
| event_type        | String | 告警类型<br>最小长度：1<br>最大长度：1024            |
| event_id          | String | 告警id<br>最小长度：1<br>最大长度：1024            |
| event_severity    | String | 告警级别<br>最小长度：1<br>最大长度：1024            |
| event_name        | String | 告警名称<br>最小长度：1<br>最大长度：1024            |
| resource_type     | String | 资源类型<br>最小长度：1<br>最大长度：1024            |
| resource_id       | String | 日志组/流名称<br>最小长度：1<br>最大长度：1024         |
| resource_provider | String | 告警源<br>最小长度：1<br>最大长度：1024             |
| lts_alarm_type    | String | 告警规则类型 (SQL/关键词)<br>最小长度：1<br>最大长度：100 |
| log_group_name    | String | 日志组原始名称                                |

| 参数              | 参数类型   | 描述      |
|-----------------|--------|---------|
| log_stream_name | String | 日志流原始名称 |

表 6-630 PageInfo

| 参数              | 参数类型   | 描述                                             |
|-----------------|--------|------------------------------------------------|
| next_marker     | String | 返回下一页查询地址（为空时，代表后面没有数据）<br>最小长度：0<br>最大长度：5000 |
| previous_marker | String | 返回前一页查询地址<br>最小长度：0<br>最大长度：5000               |
| current_count   | String | 本页返回条目数量<br>最小长度：0<br>最大长度：1000                |

## 请求示例

### 查询活动或历史告警列表

```
POST https://{endpoint}/v2/{project_id}/{domain_id}/lts/alarms/sql-alarm/query
```

```
{
 "whether_custom_field" : false,
 "start_time" : 0,
 "end_time" : 0,
 "time_range" : "30",
 "search" : "",
 "alarm_level_ids" : ["Critical", "Major", "Minor", "Info"],
 "sort" : {
 "order_by" : ["starts_at"],
 "order" : "desc"
 }
}
```

## 响应示例

### 状态码：200

请求响应成功。

```
{
 "events" : [{
 "annotations" : {
 "message" : "string",
 "log_info" : "string",
 "current_value" : "string"
 },
 "metadata" : {
 "event_type" : "string",

```

```
"event_id" : "string",
"event_severity" : "string",
"event_name" : "string",
"resource_type" : "string",
"resource_id" : "string",
"resource_provider" : "string",
"lts_alarm_type" : "string"
},
"id" : "string",
"type" : "string"
}],
"page_info" : {
"next_marker" : "string",
"previous_marker" : "string",
"current_count" : 0
}
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 查询活动或历史告警列表

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListActiveOrHistoryAlarmsSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListActiveOrHistoryAlarmsRequest request = new ListActiveOrHistoryAlarmsRequest();
 request.withDomainId("{domain_id}");
 ListActiveOrHistoryAlarmsRequestBody body = new ListActiveOrHistoryAlarmsRequestBody();
 List<String> listSortOrderBy = new ArrayList<>();
 listSortOrderBy.add("starts_at");
 Sort sortbody = new Sort();
 sortbody.withOrderBy(listSortOrderBy)
 .withOrder(Sort.OrderEnum.fromValue("desc"));
 }
}
```

```
List<String> listbodyAlarmLevelIds = new ArrayList<>();
listbodyAlarmLevelIds.add("Critical");
listbodyAlarmLevelIds.add("Major");
listbodyAlarmLevelIds.add("Minor");
listbodyAlarmLevelIds.add("Info");
body.withSort(sortbody);
body.withAlarmLevelIds(listbodyAlarmLevelIds);
body.withSearch("");
body.withTimeRange("30");
body.withEndTime(0L);
body.withStartTime(0L);
body.withWhetherCustomField(false);
request.withBody(body);
try {
 ListActiveOrHistoryAlarmsResponse response = client.listActiveOrHistoryAlarms(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 查询活动或历史告警列表

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListActiveOrHistoryAlarmsRequest()
 request.domain_id = "{domain_id}"
 listOrderBySort = [
 "starts_at"
]
 sortbody = Sort(
 order_by=listOrderBySort,
 order="desc"
)
 listAlarmLevelIdsbody = [
```

```
 "Critical",
 "Major",
 "Minor",
 "Info"
]
 request.body = ListActiveOrHistoryAlarmsRequestBody(
 sort=sortbody,
 alarm_level_ids=listAlarmLevelIdsbody,
 search="",
 time_range="30",
 end_time=0,
 start_time=0,
 whether_custom_field=False
)
 response = client.list_active_or_history_alarms(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 查询活动或历史告警列表

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListActiveOrHistoryAlarmsRequest{}
 request.DomainId = "{domain_id}"
 var listOrderBySort = []string{
 "starts_at",
 }
 sortbody := &model.Sort{
 OrderBy: listOrderBySort,
 Order: model.GetSortOrderEnum().DESC,
 }
 var listAlarmLevelIdsbody = []string{
 "Critical",
 "Major",
 "Minor",
 }
```

```
"Info",
}
searchListActiveOrHistoryAlarmsRequestBody:= ""
timeRangeListActiveOrHistoryAlarmsRequestBody:= "30"
endTimeListActiveOrHistoryAlarmsRequestBody:= int64(0)
startTimeListActiveOrHistoryAlarmsRequestBody:= int64(0)
request.Body = &model.ListActiveOrHistoryAlarmsRequestBody{
 Sort: sortbody,
 AlarmLevelIds: &listAlarmLevelIdsbody,
 Search: &searchListActiveOrHistoryAlarmsRequestBody,
 TimeRange: &timeRangeListActiveOrHistoryAlarmsRequestBody,
 EndTime: &endTimeListActiveOrHistoryAlarmsRequestBody,
 StartTime: &startTimeListActiveOrHistoryAlarmsRequestBody,
 WhetherCustomField: false,
}
response, err := client.ListActiveOrHistoryAlarms(request)
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述      |
|-----|---------|
| 200 | 请求响应成功。 |

## 错误码

请参见[错误码](#)。

## 6.14.2 删除活动告警

### 功能介绍

该接口用于删除活动告警

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/{domain\_id}/lts/alarms/sql-alarm/clear

表 6-631 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                 |
|------------|------|--------|------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |
| domain_id  | 是    | String | 账号ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a><br>最小长度: 32<br>最大长度: 32 |

## 请求参数

表 6-632 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

表 6-633 请求 Body 参数

| 参数     | 是否必选 | 参数类型                                   | 描述   |
|--------|------|----------------------------------------|------|
| events | 是    | Array of <a href="#">Event</a> objects | 主题信息 |

表 6-634 Event

| 参数        | 是否必选 | 参数类型            | 描述                             |
|-----------|------|-----------------|--------------------------------|
| metadata  | 是    | Metadata object | 告警信息<br>最小长度：0<br>最大长度：2048    |
| starts_at | 是    | Long            | 告警产生时间(时间戳)<br>最小值：0<br>最大值：32 |

表 6-635 Metadata

| 参数                | 是否必选 | 参数类型   | 描述                                    |
|-------------------|------|--------|---------------------------------------|
| event_type        | 是    | String | 告警类型<br>最小长度：1<br>最大长度：1024           |
| event_id          | 是    | String | 告警id<br>最小长度：1<br>最大长度：1024           |
| event_severity    | 是    | String | 告警级别<br>最小长度：1<br>最大长度：1024           |
| event_name        | 是    | String | 告警名称<br>最小长度：1<br>最大长度：1024           |
| resource_type     | 是    | String | 资源类型<br>最小长度：1<br>最大长度：1024           |
| resource_id       | 是    | String | 日志组/流名称<br>最小长度：1<br>最大长度：1024        |
| resource_provider | 是    | String | 告警源<br>最小长度：1<br>最大长度：1024            |
| lts_alarm_type    | 是    | String | 告警规则类型（SQL/关键词）<br>最小长度：1<br>最大长度：100 |

| 参数              | 是否必选 | 参数类型   | 描述      |
|-----------------|------|--------|---------|
| log_group_name  | 是    | String | 日志组原始名称 |
| log_stream_name | 是    | String | 日志流原始名称 |

## 响应参数

无

## 请求示例

根据告警ID删除活动告警

```
POST https://{endpoint}/v2/{project_id}/{domain_id}/lts/alarms/sql-alarm/clear
```

```
{
 "events": [{
 "metadata": {
 "event_type": "alarm",
 "event_id": "1",
 "lts_alarm_type": "keywords/sql",
 "resource_type": "日志组/流",
 "event_severity": "Critical",
 "resource_id": "lts-group-demo/lts-topic-demo",
 "event_name": "demo",
 "resource_provider": "LTS",
 "log_group_name": "lts-group-demo",
 "log_stream_name": "lts-topic-demo"
 },
 "starts_at": 1629947408497
 }
}]
}
```

## 响应示例

无

## SDK 代码示例

SDK代码示例如下。

### Java

根据告警ID删除活动告警

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;
```

```
import java.util.List;
import java.util.ArrayList;

public class DeleteActiveAlarmsSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");

 ICredential auth = new BasicCredentials()
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 DeleteActiveAlarmsRequest request = new DeleteActiveAlarmsRequest();
 request.withDomainId("{domain_id}");
 DeleteActiveAlarmsRequestBody body = new DeleteActiveAlarmsRequestBody();
 Metadata metadataEvents = new Metadata();
 metadataEvents.withEventType("alarm")
 .withEventId("1")
 .withEventSeverity("Critical")
 .withEventName("demo")
 .withResourceType("日志组/流")
 .withResourceId("lts-group-demo/lts-topic-demo")
 .withResourceProvider("LTS")
 .withLtsAlarmType("keywords/sql")
 .withLogGroupName("lts-group-demo")
 .withLogStreamName("lts-topic-demo");
 List<Event> listbodyEvents = new ArrayList<>();
 listbodyEvents.add(
 new Event()
 .withMetadata(metadataEvents)
 .withStartsAt(1629947408497L)
);
 body.withEvents(listbodyEvents);
 request.withBody(body);
 try {
 DeleteActiveAlarmsResponse response = client.deleteActiveAlarms(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

### 根据告警ID删除活动告警

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]

 credentials = BasicCredentials(ak, sk)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = DeleteActiveAlarmsRequest()
 request.domain_id = "{domain_id}"
 metadataEvents = Metadata(
 event_type="alarm",
 event_id="1",
 event_severity="Critical",
 event_name="demo",
 resource_type="日志组/流",
 resource_id="lts-group-demo/lts-topic-demo",
 resource_provider="LTS",
 lts_alarm_type="keywords/sql",
 log_group_name="lts-group-demo",
 log_stream_name="lts-topic-demo"
)
 listEventsbody = [
 Event(
 metadata=metadataEvents,
 starts_at=1629947408497
)
]
 request.body = DeleteActiveAlarmsRequestBody(
 events=listEventsbody
)
 response = client.delete_active_alarms(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 根据告警ID删除活动告警

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 Build()

client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

request := &model.DeleteActiveAlarmsRequest{}
request.DomainId = "{domain_id}"
logGroupNameMetadata:= "lts-group-demo"
logStreamNameMetadata:= "lts-topic-demo"
metadataEvents := &model.Metadata{
 EventType: "alarm",
 EventId: "1",
 EventSeverity: "Critical",
 EventName: "demo",
 ResourceType: "日志组/流",
 ResourceId: "lts-group-demo/lts-topic-demo",
 ResourceProvider: "LTS",
 LtsAlarmType: "keywords/sql",
 LogGroupName: &logGroupNameMetadata,
 LogStreamName: &logStreamNameMetadata,
}
var listEventsbody = []model.Event{
 {
 Metadata: metadataEvents,
 StartsAt: int64(1629947408497),
 },
}
request.Body = &model.DeleteActiveAlarmsRequestBody{
 Events: listEventsbody,
}
response, err := client.DeleteActiveAlarms(request)
if err == nil {
 fmt.Printf("%v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                     |
|-----|------------------------|
| 200 | 请求响应成功。                |
| 500 | 表明服务端能被请求访问到，但是服务内部出错。 |

## 错误码

请参见[错误码](#)。

## 6.15 标签管理

### 6.15.1 创建标签

#### 功能介绍

创建标签

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v1/{project\_id}/{resource\_type}/{resource\_id}/tags/action

表 6-636 路径参数

| 参数            | 是否必选 | 参数类型   | 描述                                                                            |
|---------------|------|--------|-------------------------------------------------------------------------------|
| project_id    | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| resource_type | 是    | String | 资源类型，值为groups和topics。                                                         |
| resource_id   | 是    | String | 资源id                                                                          |

#### 请求参数

表 6-637 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                          |
|--------------|------|--------|-----------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1<br>最大长度：10000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                          |
|--------------|------|--------|-------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30 |

表 6-638 请求 Body 参数

| 参数      | 是否必选 | 参数类型                             | 描述                                                                                             |
|---------|------|----------------------------------|------------------------------------------------------------------------------------------------|
| action  | 是    | String                           | 添加标签方式<br>枚举值：<br><ul style="list-style-type: none"> <li>• create</li> <li>• delete</li> </ul> |
| is_open | 是    | Boolean                          | 是否对外接口调用                                                                                       |
| tags    | 是    | Array of <b>tagsBody</b> objects | 标签字段信息                                                                                         |

表 6-639 tagsBody

| 参数    | 是否必选 | 参数类型   | 描述  |
|-------|------|--------|-----|
| key   | 否    | String | 标签键 |
| value | 否    | String | 标签值 |

## 响应参数

状态码：400

表 6-640 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：36   |
| error_msg  | String | 错误描述<br>最小长度：2<br>最大长度：512 |

**状态码：500**

**表 6-641** 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：36   |
| error_msg  | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

### 创建标签

```
/v1/2a473356cca5487f8373be891bffc1cf/groups/0933a172-379e-44d0-9ed9-f491b1c528ea/tags/action

{
 "action": "create",
 "is_open": false,
 "tags": [{
 "key": "zzz",
 "value": "zzz"
 }, {
 "key": "sgq",
 "value": "123"
 }]
}
```

## 响应示例

**状态码：400**

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "error_code": "LTS.1836",
 "error_msg": "action is create or delete"
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错

```
{
 "error_code": "LTS.0203",
 "error_msg": "Internal Server Error"
}
```

## SDK 代码示例

SDK代码示例如下。

## Java

### 创建标签

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateTagsSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateTagsRequest request = new CreateTagsRequest();
 request.withResourceType("{resource_type}");
 request.withResourceId("{resource_id}");
 CreateTagsReqbody body = new CreateTagsReqbody();
 List<TagsBody> listbodyTags = new ArrayList<>();
 listbodyTags.add(
 new TagsBody()
 .withKey("zzz")
 .withValue("zzz")
);
 listbodyTags.add(
 new TagsBody()
 .withKey("sgq")
 .withValue("123")
);
 body.withTags(listbodyTags);
 body.withIsOpen(false);
 body.withAction(CreateTagsReqbody.ActionEnum.fromValue("create"));
 request.withBody(body);
 try {
 CreateTagsResponse response = client.createTags(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 }
 }
}
```

```
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 创建标签

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateTagsRequest()
 request.resource_type = "{resource_type}"
 request.resource_id = "{resource_id}"
 listTagsbody = [
 TagsBody(
 key="zzz",
 value="zzz"
),
 TagsBody(
 key="sgq",
 value="123"
)
]
 request.body = CreateTagsReqbody(
 tags=listTagsbody,
 is_open=False,
 action="create"
)
 response = client.create_tags(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建标签

```
package main
```

```
import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateTagsRequest{
 request.ResourceType = "{resource_type}"
 request.ResourceId = "{resource_id}"
 keyTags:= "zzz"
 valueTags:= "zzz"
 keyTags1:= "sgq"
 valueTags1:= "123"
 var listTagsbody = []model.TagsBody{
 {
 Key: &keyTags,
 Value: &valueTags,
 },
 {
 Key: &keyTags1,
 Value: &valueTags1,
 },
 }
 }
 request.Body = &model.CreateTagsReqbody{
 Tags: listTagsbody,
 IsOpen: false,
 Action: model.GetCreateTagsReqbodyActionEnum().CREATE,
 }
 response, err := client.CreateTags(request)
 if err == nil {
 fmt.Printf("%v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 201 | 添加标签成功                              |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.16 仪表盘管理

### 6.16.1 创建仪表盘分组

#### 功能介绍

创建仪表盘分组

#### 调用方法

请参见[如何调用API](#)。

#### URI

POST /v2/{project\_id}/lts/dashboard-group

表 6-642 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

## 请求参数

表 6-643 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                              |
|--------------|------|--------|---------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，<br>获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1<br>最大长度：10000 |
| Content-Type | 是    | String | 该字段填为：application/<br>json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                 |

表 6-644 请求 Body 参数

| 参数         | 是否必选 | 参数类型   | 描述      |
|------------|------|--------|---------|
| group_name | 是    | String | 仪表盘分组名称 |

## 响应参数

状态码：201

表 6-645 响应 Body 参数

| 参数     | 参数类型   | 描述   |
|--------|--------|------|
| result | String | 响应结果 |

状态码：400

表 6-646 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-647 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-648 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-649 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

创建仪表盘分组

```
/v2/2a473356cca5487f8373be891bffc1cf/lts/dashboard-group
```

```
{
 "group_name": "sgqtest"
}
```

## 响应示例

状态码：201

创建仪表盘分组成功

```
{
 "result" : "success"
}
```

**状态码：400**

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message" : {
 "code" : "LTS.2111",
 "details" : "Log dashboard group name is already exist"
 }
}
```

**状态码：500**

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 创建仪表盘分组

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class CreateDashboardGroupSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
```

```
CreateDashboardGroupRequest request = new CreateDashboardGroupRequest();
CreateDashboardGroupReq body = new CreateDashboardGroupReq();
body.withGroupName("sgqtest");
request.withBody(body);
try {
 CreateDashboardGroupResponse response = client.createDashboardGroup(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

### 创建仪表盘分组

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateDashboardGroupRequest()
 request.body = CreateDashboardGroupReq(
 group_name="sgqtest"
)
 response = client.create_dashboard_group(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 创建仪表盘分组

```
package main
```

```
import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateDashboardGroupRequest{
 request.Body = &model.CreateDashboardGroupReq{
 GroupName: "sgqtest",
 }
 }
 response, err := client.CreateDashboardGroup(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 201 | 创建仪表盘分组成功                           |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.16.2 创建仪表盘

### 功能介绍

创建仪表盘

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v2/{project\_id}/dashboard

表 6-650 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

### 请求参数

表 6-651 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                          |
|--------------|------|--------|-----------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1<br>最大长度：10000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                 |

表 6-652 请求 Body 参数

| 参数         | 是否必选 | 参数类型   | 描述      |
|------------|------|--------|---------|
| group_name | 否    | String | 仪表盘分组名称 |

| 参数    | 是否必选 | 参数类型   | 描述    |
|-------|------|--------|-------|
| title | 是    | String | 仪表盘名称 |

## 响应参数

状态码：201

表 6-653 响应 Body 参数

| 参数                  | 参数类型                                         | 描述     |
|---------------------|----------------------------------------------|--------|
| charts              | Array of strings                             | 仪表盘图表  |
| filters             | Array of strings                             | 过滤条件   |
| group_name          | String                                       | 日志组名称  |
| id                  | String                                       | 仪表盘id  |
| last_update_time    | Long                                         | 最近修改时间 |
| project_id          | String                                       | 项目id   |
| title               | String                                       | 仪表盘名称  |
| useSystemTemplate   | Boolean                                      | 是否使用模板 |
| tags                | Array of <a href="#">ResourceTag</a> objects | 仪表盘标签  |
| concise_mode_enable | String                                       | 纯净模式开关 |

表 6-654 ResourceTag

| 参数                     | 参数类型   | 描述        |
|------------------------|--------|-----------|
| key                    | String | 标签的key    |
| value                  | String | 标签的value  |
| tags_to_streams_enable | String | 是否启用日志流标签 |

状态码：400

表 6-655 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-656 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-657 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-658 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

创建仪表盘，group\_name为非必填参数。

```
/v2/2a473356cca5487f8373be891bffc1cf/dashboard
```

```
{
 "group_name": "sgqtest",
 "title": "sgqtest1"
}
```

## 响应示例

### 状态码：201

创建仪表盘成功

```
{
 "charts": [],
 "filters": [],
 "group_name": "",
 "id": "2bf23310-83c2-4962-898f-79ebd057a256",
 "last_update_time": 1669637866189,
 "project_id": "2a473356cca5487f8373be891bffc1cf",
 "title": "test",
 "useSystemTemplate": false
}
```

### 状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message": {
 "code": "LTS.0736",
 "details": "The number of log dashboards exceeds the maximum"
 }
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message": {
 "code": "LTS.0203",
 "details": "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

创建仪表盘，group\_name为非必填参数。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class CreateDashBoardSolution {
```

```
public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateDashBoardRequest request = new CreateDashBoardRequest();
 CreateDashBoardReqBody body = new CreateDashBoardReqBody();
 body.withTitle("sgqtest1");
 body.withGroupName("sgqtest");
 request.withBody(body);
 try {
 CreateDashBoardResponse response = client.createDashBoard(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
```

## Python

创建仪表盘，group\_name为非必填参数。

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()
```

```
try:
 request = CreateDashBoardRequest()
 request.body = CreateDashBoardReqBody(
 title="sgqtest1",
 group_name="sgqtest"
)
 response = client.create_dash_board(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

创建仪表盘，group\_name为非必填参数。

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateDashBoardRequest{
 groupNameCreateDashBoardReqBody:= "sgqtest"
 }
 request.Body = &model.CreateDashBoardReqBody{
 Title: "sgqtest1",
 GroupName: &groupNameCreateDashBoardReqBody,
 }
 response, err := client.CreateDashBoard(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 201 | 创建仪表盘成功                             |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.17 日志流图表

### 6.17.1 查询日志流图表

#### 功能介绍

该接口用于查询日志流图表

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v2/{project\_id}/groups/{log\_group\_id}/streams/{log\_stream\_id}/charts

表 6-659 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

| 参数            | 是否必选 | 参数类型   | 描述                                                                                                                                                   |
|---------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| log_group_id  | 是    | String | 日志组ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度： <b>36</b><br>最大长度： <b>36</b> |
| log_stream_id | 是    | String | 日志流ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a><br>最小长度： <b>36</b><br>最大长度： <b>36</b> |

表 6-660 Query 参数

| 参数     | 是否必选 | 参数类型    | 描述                                                             |
|--------|------|---------|----------------------------------------------------------------|
| offset | 否    | Integer | 查询游标，初始传入0，后续从上一次的返回值中获取。<br>最小值： <b>0</b><br>最大值： <b>1024</b> |
| limit  | 否    | Integer | 每页数据量，最大值为100。<br>最小值： <b>0</b><br>最大值： <b>100</b>             |

## 请求参数

表 6-661 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                            |
|--------------|------|--------|-----------------------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度： <b>1000</b><br>最大长度： <b>2000</b> |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度： <b>30</b><br>最大长度： <b>30</b>                   |

## 响应参数

状态码：200

表 6-662 响应 Body 参数

| 参数              | 参数类型                         | 描述                                                                                                                               |
|-----------------|------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| id              | String                       | id<br>最小长度：36<br>最大长度：36                                                                                                         |
| sql             | String                       | sql语句<br>最小长度：1<br>最大长度：1024                                                                                                     |
| title           | String                       | 图表名称<br>最小长度：1<br>最大长度：64                                                                                                        |
| type            | String                       | 图表类型<br>枚举值： <ul style="list-style-type: none"><li>• table</li><li>• bar</li><li>• line</li><li>• pie</li><li>• number</li></ul> |
| log_group_id    | String                       | 日志组id<br>最小长度：36<br>最大长度：36                                                                                                      |
| log_group_name  | String                       | 日志组名称<br>最小长度：1<br>最大长度：64                                                                                                       |
| log_stream_id   | String                       | 日志组id<br>最小长度：32<br>最大长度：32                                                                                                      |
| log_stream_name | String                       | 日志流名称<br>最小长度：1<br>最大长度：64                                                                                                       |
| config          | <b>ChartConfig</b><br>object | 图表配置详情                                                                                                                           |

表 6-663 ChartConfig

| 参数        | 参数类型    | 描述                         |
|-----------|---------|----------------------------|
| canSort   | Boolean | 是否开启排序                     |
| canSearch | Boolean | 是否开启搜索                     |
| pageSize  | Integer | 每页显示数量<br>最小值：0<br>最大值：100 |

## 请求示例

查询日志流图表

```
GET https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/charts
/v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/charts
```

## 响应示例

状态码：200

请求响应成功。

```
{
 "id": "string",
 "sql": "string",
 "title": "string",
 "type": "table",
 "log_group_id": "string",
 "log_group_name": "string",
 "log_stream_id": "string",
 "log_stream_name": "string",
 "config": {
 "canSort": true,
 "canSearch": true,
 "pageSize": 0
 }
}
```

## 状态码

| 状态码 | 描述      |
|-----|---------|
| 200 | 请求响应成功。 |

## 错误码

请参见[错误码](#)。

## 6.18 快速查询

## 6.18.1 查询用户历史 sql

### 功能介绍

查询用户历史sql

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v2/{project\_id}/lts/history-sql

表 6-664 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |

表 6-665 Query 参数

| 参数            | 是否必选 | 参数类型   | 描述                                                                             |
|---------------|------|--------|--------------------------------------------------------------------------------|
| log_group_id  | 是    | String | 日志组ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36 |
| log_stream_id | 是    | String | 日志流ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：36<br>最大长度：36 |

## 请求参数

表 6-666 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                          |
|--------------|------|--------|-----------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1<br>最大长度：10000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                 |

## 响应参数

状态码：200

表 6-667 响应 Body 参数

| 参数      | 参数类型                                                        | 描述    |
|---------|-------------------------------------------------------------|-------|
| results | Array of <a href="#">QuertHistorySQLResultsBody</a> objects | 响应结果。 |

表 6-668 QuertHistorySQLResultsBody

| 参数            | 参数类型   | 描述             |
|---------------|--------|----------------|
| last_use_time | Long   | 上次修改时间，时间戳，毫秒数 |
| sql_statement | String | 历史sql语句        |

状态码：400

表 6-669 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-670 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-671 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-672 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

查询用户历史sql

```
/v2/2a473356cca5487f8373be891bffc1cf/lts/history-sql?log_group_id=d1f4240d-5ee2-4e0b-9e2c-e25c7978c001&log_stream_id=f3d853be-0576-4dff-ae9-c093e4924b63
```

## 响应示例

状态码：200

查询sql历史成功

```
{
 "results": [{
 "last_use_time": 1669689650936,
 }]
```

```
"sql_statement" : "SELECT *"
}, {
 "last_use_time" : 1666775421613,
 "sql_statement" : "select count(\"t\") as pv"
}, {
 "last_use_time" : 1666661494805,
 "sql_statement" : "SELECT count(\"time\")"
}, {
 "last_use_time" : 1666598233252,
 "sql_statement" : "SELECT count(\"t\") as pv"
}, {
 "last_use_time" : 1666598226763,
 "sql_statement" : "SELECT count(\"t\") "
}, {
 "last_use_time" : 1666598222298,
 "sql_statement" : "SELECT count(\"t\") as pv"
}, {
 "last_use_time" : 1666598221585,
 "sql_statement" : "SELECT count(\"t\") as p"
}, {
 "last_use_time" : 1666598220276,
 "sql_statement" : "SELECT count(\"t\") as "
}, {
 "last_use_time" : 1666598212453,
 "sql_statement" : "SELECT count(\"t\")"
}]
}
```

#### 状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message" : {
 "code" : "LTS.0603",
 "details" : "group or stream not exist"
 }
}
```

#### 状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;
```

```
public class ListHistorySqlSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 ListHistorySqlRequest request = new ListHistorySqlRequest();
 try {
 ListHistorySqlResponse response = client.listHistorySql(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
 }
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.getenv("CLOUD_SDK_AK")
 sk = os.getenv("CLOUD_SDK_SK")
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListHistorySqlRequest()
 response = client.list_history_sql(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListHistorySqlRequest{}
 response, err := client.ListHistorySql(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 200 | 查询sql历史成功                           |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.18.2 添加快速查询

### 功能介绍

添加快速查询

### 调用方法

请参见[如何调用API](#)。

### URI

POST /v1.0/{project\_id}/groups/{group\_id}/topics/{topic\_id}/search-criterias

表 6-673 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| group_id   | 是    | String | 租户想查询的日志流所在的日志组的groupid，一般为36位字符串。<br>最小长度：36<br>最大长度：36                      |
| topic_id   | 是    | String | 日志流id                                                                         |

### 请求参数

表 6-674 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |

| 参数           | 是否必选 | 参数类型   | 描述                                                                      |
|--------------|------|--------|-------------------------------------------------------------------------|
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>缺省值：None<br>最小长度：30<br>最大长度：30 |

表 6-675 请求 Body 参数

| 参数          | 是否必选 | 参数类型   | 描述                                               |
|-------------|------|--------|--------------------------------------------------|
| criteria    | 是    | String | 快速查询字段                                           |
| eps_id      | 否    | String | 企业项目id                                           |
| name        | 是    | String | 创建快速查询名称                                         |
| search_type | 是    | String | 查询类型原始日志：<br>ORIGINALLOG<br>可视化日志: VISUALIZATION |

## 响应参数

状态码：201

表 6-676 响应 Body 参数

| 参数 | 参数类型   | 描述     |
|----|--------|--------|
| id | String | 快速查询id |

状态码：400

表 6-677 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-678 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-679 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-680 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

添加快速查询

```
/v1.0/2a473356cca5487f8373be891bffc1cf/groups/d1f4240d-5ee2-4e0b-9e2c-e25c7978c001/topics/2b899d46-218c-4f0c-8ace-a36a290a83a0/search-criterias
```

```
{
 "name": "创建数字",
 "criteria": "content : 1234567891234567891234567891234567891234567891234567891234567894",
 "eps_id": "0",
 "search_type": "ORIGINALLOG"
}
```

## 响应示例

### 状态码：201

添加快速查询成功

```
{
 "id" : "0eb379f5-f847-4d25-ba89-05967bf1bae3"
}
```

### 状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message" : {
 "code" : "LTS.0208",
 "details" : "The log stream does not existed"
 }
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

添加快速查询

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class CreateSearchCriteriaSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
```

```
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 CreateSearchCriteriaRequest request = new CreateSearchCriteriaRequest();
 request.withGroupId("{group_id}");
 request.withTopicId("{topic_id}");
 CreateSearchCriteriaBody body = new CreateSearchCriteriaBody();
 body.withSearchType("ORIGINALLOG");
 body.withName("创建数字");
 body.withEpsId("0");
 body.withCriteria("content :
1234567891234567891234567891234567891234567891234567891234567891234567891234567891234567894");
 request.withBody(body);
 try {
 CreateSearchCriteriaResponse response = client.createSearchCriteria(request);
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrMsg());
 }
}
```

## Python

### 添加快速查询

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = CreateSearchCriteriaRequest()
 request.group_id = "{group_id}"
 request.topic_id = "{topic_id}"
 request.body = CreateSearchCriteriaBody(
 search_type="ORIGINALLOG",
```

```
 name="创建数字",
 eps_id="0",
 criteria="content : 1234567891234567891234567891234567891234567891234567891234567891234567894"
)
 response = client.create_search_criterias(request)
 print(response)
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 添加快速查询

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.CreateSearchCriteriasRequest{}
 request.GroupId = "{group_id}"
 request.TopicId = "{topic_id}"
 epsIdCreateSearchCriteriasBody := "0"
 request.Body = &model.CreateSearchCriteriasBody{
 SearchType: "ORIGINALLOG",
 Name: "创建数字",
 EpsId: &epsIdCreateSearchCriteriasBody,
 Criteria: "content : 1234567891234567891234567891234567891234567891234567891234567894",
 }
 response, err := client.CreateSearchCriterias(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 201 | 添加快速查询成功                            |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

### 6.18.3 获取快速查询

#### 功能介绍

获取快速查询

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v1.0/{project\_id}/groups/{group\_id}/topics/{topic\_id}/search-criterias

表 6-681 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| group_id   | 是    | String | 租户想查询的日志流所在的日志组的groupid，一般为36位字符串。<br>最小长度：36<br>最大长度：36                      |
| topic_id   | 是    | String | 日志流id                                                                         |

表 6-682 Query 参数

| 参数          | 是否必选 | 参数类型   | 描述                                        |
|-------------|------|--------|-------------------------------------------|
| search_type | 否    | String | 原始日志: ORIGINALLOG<br>可视化日志: VISUALIZATION |

## 请求参数

表 6-683 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                               |
|--------------|------|--------|----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a><br>最小长度: 1000<br>最大长度: 2000 |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。<br>最小长度: 30<br>最大长度: 30                   |

## 响应参数

状态码: 200

表 6-684 响应 Body 参数

| 参数               | 参数类型                                                         | 描述       |
|------------------|--------------------------------------------------------------|----------|
| search_criterias | Array of <a href="#">GetQuerySearchCriteriasBody</a> objects | 响应Body体。 |

表 6-685 GetQuerySearchCriteriasBody

| 参数       | 参数类型   | 描述     |
|----------|--------|--------|
| criteria | String | 快速查询字段 |
| name     | String | 快速查询名称 |
| id       | String | 快速查询id |

| 参数          | 参数类型   | 描述                                                   |
|-------------|--------|------------------------------------------------------|
| search_type | String | 快速查询类型。<br>原始日志: ORIGINALLOG<br>可视化日志: VISUALIZATION |

状态码: 400

表 6-686 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-687 ErrorMessagebody

| 参数      | 参数类型   | 描述                           |
|---------|--------|------------------------------|
| code    | String | 错误码<br>最小长度: 8<br>最大长度: 36   |
| details | String | 错误描述<br>最小长度: 2<br>最大长度: 512 |

状态码: 500

表 6-688 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-689 ErrorMessagebody

| 参数   | 参数类型   | 描述                         |
|------|--------|----------------------------|
| code | String | 错误码<br>最小长度: 8<br>最大长度: 36 |

| 参数      | 参数类型   | 描述                           |
|---------|--------|------------------------------|
| details | String | 错误描述<br>最小长度: 2<br>最大长度: 512 |

## 请求示例

获取快速查询

```
/v1.0/2a473356cca5487f8373be891bffc1cf/groups/d1f4240d-5ee2-4e0b-9e2c-e25c7978c001/topics/
2b899d46-218c-4f0c-8ace-a36a290a83a0/search-criterias?search_type=ORIGINALLOG

search_type:
ORIGINALLOG
```

## 响应示例

**状态码: 200**

获取快速查询成功

```
{
 "search_criterias": [{
 "criteria": "content : 1234567891234567891234567891234567891234567891234567891234567894",
 "name": "创建数字",
 "id": "0eb379f5-f847-4d25-ba89-05967bf1bae3",
 "search_type": "ORIGINALLOG"
 }]
}
```

**状态码: 400**

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message": {
 "code": "LTS.0208",
 "details": "The log stream does not existed"
 }
}
```

**状态码: 500**

表明服务端能被请求访问到, 但是服务内部出错

```
{
 "error_code": "LTS.0203",
 "error_msg": "Internal Server Error"
}
```

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 200 | 获取快速查询成功                            |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |

| 状态码 | 描述                    |
|-----|-----------------------|
| 500 | 表明服务端能被请求访问到，但是服务内部出错 |

## 错误码

请参见[错误码](#)。

## 6.18.4 删除快速查询

### 功能介绍

删除快速查询

### 调用方法

请参见[如何调用API](#)。

### URI

DELETE /v1.0/{project\_id}/groups/{group\_id}/topics/{topic\_id}/search-criterias

表 6-690 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| group_id   | 是    | String | 租户想查询的日志流所在的日志组的groupid，一般为36位字符串。<br>最小长度：36<br>最大长度：36                      |
| topic_id   | 是    | String | 日志流id                                                                         |

## 请求参数

表 6-691 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                                |
|--------------|------|--------|-----------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，<br>获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/<br>json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

表 6-692 请求 Body 参数

| 参数     | 是否必选 | 参数类型   | 描述     |
|--------|------|--------|--------|
| eps_id | 否    | String | 企业项目id |
| id     | 是    | String | 快速查询id |

## 响应参数

状态码：400

表 6-693 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-694 ErrorMessagebody

| 参数   | 参数类型   | 描述                       |
|------|--------|--------------------------|
| code | String | 错误码<br>最小长度：8<br>最大长度：36 |

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-695 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-696 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

查询快速查询

```
/v1.0/2a473356cca5487f8373be891bffc1cf/groups/d1f4240d-5ee2-4e0b-9e2c-e25c7978c001/topics/2b899d46-218c-4f0c-8ace-a36a290a83a0/search-criterias
```

```
{
 "id": "345d2276-1ae8-4495-a6ee-bf77c2e5ffb9",
 "epsId": "0"
}
```

## 响应示例

状态码：204

删除快速查询成功

状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
 "message" : {
 "code" : "LTS.0208",
 "details" : "The log stream does not existed"
 }
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
 "message" : {
 "code" : "LTS.0203",
 "details" : "Internal Server Error"
 }
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

#### 查询快速查询

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class DeleteSearchCriteriaSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 DeleteSearchCriteriaRequest request = new DeleteSearchCriteriaRequest();
 request.withGroupId("{group_id}");
 request.withTopicId("{topic_id}");
 DeleteSearchCriteria body = new DeleteSearchCriteria();
 body.withId("345d2276-1ae8-4495-a6ee-bf77c2e5ffb9");
 request.withBody(body);
 try {
 DeleteSearchCriteriaResponse response = client.deleteSearchCriteria(request);
 }
 }
}
```

```
 System.out.println(response.toString());
 } catch (ConnectionException e) {
 e.printStackTrace();
 } catch (RequestTimeoutException e) {
 e.printStackTrace();
 } catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
 }
}
}
```

## Python

### 查询快速查询

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = DeleteSearchCriteriaRequest()
 request.group_id = "{group_id}"
 request.topic_id = "{topic_id}"
 request.body = DeleteSearchCriteria(
 id="345d2276-1ae8-4495-a6ee-bf77c2e5ffb9"
)
 response = client.delete_search_criteria(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

### 查询快速查询

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.DeleteSearchCriteriaRequest{}
 request.GroupId = "{group_id}"
 request.TopicId = "{topic_id}"
 request.Body = &model.DeleteSearchCriteria{
 Id: "345d2276-1ae8-4495-a6ee-bf77c2e5ffb9",
 }
 response, err := client.DeleteSearchCriteria(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 204 | 删除快速查询成功                            |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.18.5 查询日志组下所有快速查询

### 功能介绍

查询日志组下所有快速查询

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v1.0/{project\_id}/lts/groups/{group\_id}/search-criterias

表 6-697 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                            |
|------------|------|--------|-------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a><br>最小长度：32<br>最大长度：32 |
| group_id   | 是    | String | 待查询的日志流所在的日志组的groupid，一般为36位字符串。<br>最小长度：36<br>最大长度：36                        |

### 请求参数

表 6-698 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                                            |
|--------------|------|--------|-------------------------------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a><br>最小长度：1000<br>最大长度：2000 |
| Content-Type | 是    | String | 该字段填为：application/json;charset=UTF-8。<br>最小长度：30<br>最大长度：30                   |

## 响应参数

状态码：200

表 6-699 响应 Body 参数

| 参数               | 参数类型                                                  | 描述   |
|------------------|-------------------------------------------------------|------|
| search_criterias | Array of <a href="#">search_criteriasBody</a> objects | 快速查询 |

表 6-700 search\_criteriasBody

| 参数              | 参数类型                                                         | 描述         |
|-----------------|--------------------------------------------------------------|------------|
| criterias       | Array of <a href="#">GetQuerySearchCriteriasBody</a> objects | 单个日志流的快速查询 |
| log_stream_id   | String                                                       | 日志流id      |
| log_stream_name | String                                                       | 日志流名称      |

表 6-701 GetQuerySearchCriteriasBody

| 参数          | 参数类型   | 描述                                                   |
|-------------|--------|------------------------------------------------------|
| criteria    | String | 快速查询字段                                               |
| name        | String | 快速查询名称                                               |
| id          | String | 快速查询id                                               |
| search_type | String | 快速查询类型。<br>原始日志: ORIGINALLOG<br>可视化日志: VISUALIZATION |

状态码：400

表 6-702 响应 Body 参数

| 参数      | 参数类型                                    | 描述        |
|---------|-----------------------------------------|-----------|
| message | <a href="#">ErrorMessagebody</a> object | 错误信息body体 |

表 6-703 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

状态码：500

表 6-704 响应 Body 参数

| 参数      | 参数类型                           | 描述        |
|---------|--------------------------------|-----------|
| message | <b>ErrorMessagebody</b> object | 错误信息body体 |

表 6-705 ErrorMessagebody

| 参数      | 参数类型   | 描述                         |
|---------|--------|----------------------------|
| code    | String | 错误码<br>最小长度：8<br>最大长度：36   |
| details | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

查询日志组下所有快速查询

```
/v1.0/2a473356cca5487f8373be891bffc1cf/lts/groups/d1f4240d-5ee2-4e0b-9e2c-e25c7978c001/search-criterias
```

## 响应示例

状态码：200

查询成功

```
{
 "search_criterias": [{
 "criterias": [{
```

```
"criteria": "234",
"name": "234",
"id": "aee53496-xxxx-xxxx-xxxx-5a83d1aae134",
"search_type": "ORIGINALLOG"
}, {
"criteria": "24",
"name": "2342",
"id": "939c930a-xxxx-xxxx-xxxx-a3040d857ce1",
"search_type": "ORIGINALLOG"
}],
"log_stream_id": "85025a4b-xxxx-xxxx-xxxx-2b6851d84ea2",
"log_stream_name": "lts-test-topic",
"search_type": "ORIGINALLOG"
}]
}
```

### 状态码：400

BadRequest 非法请求建议根据error\_msg直接修改该请求

```
{
"message": {
"code": "LTS.0201",
"details": "The log group does not existed"
}
}
```

### 状态码：500

表明服务端能被请求访问到，但是服务内部出错

```
{
"message": {
"code": "LTS.0203",
"details": "Internal Server Error"
}
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

public class ListQueryAllSearchCriteriaSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
ListQueryAllSearchCriteriaRequest request = new ListQueryAllSearchCriteriaRequest();
request.withGroupId("{group_id}");
try {
 ListQueryAllSearchCriteriaResponse response = client.listQueryAllSearchCriteria(request);
 System.out.println(response.toString());
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudskcls.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudskcls.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = ListQueryAllSearchCriteriaRequest()
 request.group_id = "{group_id}"
 response = client.list_query_all_search_criteria(request)
 print(response)
 except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.ListQueryAllSearchCriteriaRequest{}
 request.GroupId = "{group_id}"
 response, err := client.ListQueryAllSearchCriteria(request)
 if err == nil {
 fmt.Printf("%+v\n", response)
 } else {
 fmt.Println(err)
 }
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述                                  |
|-----|-------------------------------------|
| 200 | 查询成功                                |
| 400 | BadRequest 非法请求建议根据error_msg直接修改该请求 |
| 500 | 表明服务端能被请求访问到，但是服务内部出错               |

## 错误码

请参见[错误码](#)。

## 6.19 多账号日志汇聚

### 6.19.1 获取日志汇聚开关

#### 功能介绍

只能由管理员或者委托管理员调用，获取日志汇聚开关

#### 调用方法

请参见[如何调用API](#)。

#### URI

GET /v1/{project\_id}/lts/log-converge-config/switch

表 6-706 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                      |
|------------|------|--------|---------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID，获取账号ID，日志组ID、日志流ID</a> |

#### 请求参数

表 6-707 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                  |
|--------------|------|--------|-----------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为：application/json;charset=utf8。                |

#### 响应参数

状态码：200

表 6-708 响应 Body 参数

| 参数                  | 参数类型    | 描述     |
|---------------------|---------|--------|
| log_converge_switch | Boolean | 日志汇聚开关 |

状态码：400

表 6-709 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：36   |
| error_msg  | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

获取日志汇聚开关

```
GET https://{endpoint}/v1/{project_id}/lts/log-converge-config/switch
/v1/{project_id}/lts/log-converge-config/switch
```

## 响应示例

状态码：200

查询成功

```
{
 "log_converge_switch" : true
}
```

## 状态码

| 状态码 | 描述             |
|-----|----------------|
| 200 | 查询成功           |
| 400 | Error response |

## 错误码

请参见[错误码](#)。

## 6.19.2 修改日志汇聚开关

### 功能介绍

只能由管理员或者委托管理员调用 修改日志汇聚开关

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v1/{project\_id}/lts/log-converge-config/switch

表 6-710 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                                                                            |
|------------|------|--------|---------------------------------------------------------------------------------------------------------------|
| project_id | 是    | String | 项目ID，获取方式请参见： <a href="#">获取项目ID</a> ， <a href="#">获取账号ID</a> ， <a href="#">日志组ID</a> 、 <a href="#">日志流ID</a> |

表 6-711 Query 参数

| 参数                  | 是否必选 | 参数类型   | 描述   |
|---------------------|------|--------|------|
| log_converge_switch | 是    | String | 开关参数 |

### 请求参数

表 6-712 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                  |
|--------------|------|--------|-----------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token，获取方式请参见： <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为：application/json;charset=utf8。                |

### 响应参数

状态码：200

表 6-713 响应 Body 参数

| 参数     | 参数类型   | 描述       |
|--------|--------|----------|
| result | String | 修改日志汇聚结果 |

状态码：400

表 6-714 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：36   |
| error_msg  | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

日志汇聚开关打开

```
PUT https://{endpoint}/v1/{project_id}/lts/log-converge-config/switch
/v1/{project_id}/lts/log-converge-config/switch?log_converge_switch=true
```

## 响应示例

状态码：200

修改日志汇聚响应体

```
{
 "result" : "success"
}
```

## 状态码

| 状态码 | 描述             |
|-----|----------------|
| 200 | 修改日志汇聚响应体      |
| 400 | Error response |

## 错误码

请参见[错误码](#)。

## 6.19.3 获取组织成员汇聚配置

### 功能介绍

只能由组织管理员或者委托管理员调用 获取组织成员汇聚配置

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v1/{project\_id}/lts/log-converge-config/{member\_account\_id}

表 6-715 路径参数

| 参数                | 是否必选 | 参数类型   | 描述                                                         |
|-------------------|------|--------|------------------------------------------------------------|
| project_id        | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a> |
| member_account_id | 是    | String | 成员账户ID                                                     |

### 请求参数

表 6-716 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                   |
|--------------|------|--------|------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。               |

### 响应参数

状态码: 200

表 6-717 响应 Body 参数

| 参数 | 参数类型   | 描述 |
|----|--------|----|
| id | String | ID |

| 参数                    | 参数类型                                              | 描述                                                                                                                           |
|-----------------------|---------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| member_account_id     | String                                            | 组织成员账号id                                                                                                                     |
| member_project_id     | String                                            | 管理员或者委托管理员项目id                                                                                                               |
| create_time           | Long                                              | 创建时间                                                                                                                         |
| update_time           | Long                                              | 更新时间                                                                                                                         |
| status                | String                                            | creating: 配置创建中 done: 配置创建完成<br>枚举值: <ul style="list-style-type: none"><li>• <b>creating</b></li><li>• <b>done</b></li></ul> |
| organization_id       | String                                            | 组织id                                                                                                                         |
| management_account_id | String                                            | 管理员或者委托管理员账号id                                                                                                               |
| management_project_id | String                                            | 管理员项目id                                                                                                                      |
| version               | String                                            | 版本                                                                                                                           |
| log_mapping_config    | Array of <a href="#">LogMappingConfig</a> objects | 日志汇聚配置                                                                                                                       |

表 6-718 LogMappingConfig

| 参数                    | 参数类型                                                  | 描述      |
|-----------------------|-------------------------------------------------------|---------|
| source_log_group_id   | String                                                | 源日志组ID  |
| target_log_group_id   | String                                                | 目标日志组ID |
| target_log_group_name | String                                                | 目标日志组名称 |
| log_stream_config     | Array of <a href="#">LogMappingStreamInfo</a> objects | 日志流配置   |

表 6-719 LogMappingStreamInfo

| 参数                       | 参数类型    | 描述                                         |
|--------------------------|---------|--------------------------------------------|
| source_log_stream_id     | String  | 源日志流ID                                     |
| target_log_stream_id     | String  | 目标日志流ID                                    |
| target_log_stream_name   | String  | 目标日志流名称                                    |
| target_log_stream_eps_id | String  | 目标日志流EPS ID                                |
| target_log_stream_ttl    | Integer | 目标日志流ttl保存时间,单位时间(天).<br>最小值: 1<br>最大值: 30 |

状态码: 400

表 6-720 响应 Body 参数

| 参数         | 参数类型   | 描述                           |
|------------|--------|------------------------------|
| error_code | String | 错误码<br>最小长度: 8<br>最大长度: 36   |
| error_msg  | String | 错误描述<br>最小长度: 2<br>最大长度: 512 |

## 请求示例

获取组织成员汇聚配置

```
GET https://{endpoint}/v1/{project_id}/lts/log-converge-config/{member_account_id}
/v1/{project_id}/lts/log-converge-config/{member_account_id}
```

## 响应示例

状态码: 200

Delete result

```
{
 "create_time": 1694423493108,
 "id": "f5b6b235-48a2-473c-a1fa-4ddba9ca86d8",
 "log_mapping_config": [{
 "log_stream_config": [{
```

```
"source_log_stream_id": "55b33967-1971-4521-bf5a-e53e981f3d08",
"target_log_stream_eps_id": "0",
"target_log_stream_id": "ebaf8e21-dba6-4697-99b2-6811f109d03f",
"target_log_stream_name": "stdout-test_huiju",
"target_log_stream_ttl": 180
}],
"source_log_group_id": "c59e2705-5bbf-4f55-8861-92d4e7e7d33f",
"target_log_group_id": "5adb3025-e663-4cf4-bcf7-7340e79cddb3",
"target_log_group_name": "k8s-log-test_paas_apm_z0041xxxx_01"
}, {
"log_stream_config": [{
"source_log_stream_id": "b77c9164-b411-42f4-9d4c-2ca7115496b7",
"target_log_stream_eps_id": "0",
"target_log_stream_id": "7120856a-bf15-4f92-8f08-b64bba8c81b1",
"target_log_stream_name": "stdout-test_huiju",
"target_log_stream_ttl": 180
}],
"source_log_group_id": "77f7e0fc-0ad0-4af0-b1e9-198007d5f993",
"target_log_group_id": "f8076177-c903-4dcf-85d5-dca9d13e591e",
"target_log_group_name": "k8s-log-stest_paas_apm_z0041xxxx_01"
}],
"management_account_id": "115ba1c5fec44839820795ebefe25f2a",
"management_project_id": "a5baef478e6840dc9f454bbc00c996c6",
"member_account_id": "1d26cc8c86a840e28a4f8d0d07852f1d",
"member_project_id": "2a473356cca5487f8373be891bffc1cf",
"organization_id": "o-fa9h4ghf0hnpfnq4xd02tah2nxmtugrf",
"status": "done"
}
```

**状态码：400**

Error response

""

## 状态码

| 状态码 | 描述             |
|-----|----------------|
| 200 | Delete result  |
| 400 | Error response |

## 错误码

请参见[错误码](#)。

## 6.19.4 获取组织成员日志组日志流

### 功能介绍

只能由管理员或者委托管理员调用，获取组织成员日志组日志流

### 调用方法

请参见[如何调用API](#)。

### URI

GET /v1/{project\_id}/lts/{member\_account\_id}/all-streams

表 6-721 路径参数

| 参数                | 是否必选 | 参数类型   | 描述                                                         |
|-------------------|------|--------|------------------------------------------------------------|
| project_id        | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a> |
| member_account_id | 是    | String | 成员账号ID                                                     |

## 请求参数

表 6-722 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                   |
|--------------|------|--------|------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。               |

## 响应参数

状态码: 200

表 6-723 响应 Body 参数

| 参数      | 参数类型                                     | 描述             |
|---------|------------------------------------------|----------------|
| results | Array of <a href="#">results</a> objects | 获取组织成员日志组日志流结果 |

表 6-724 results

| 参数             | 参数类型                                         | 描述    |
|----------------|----------------------------------------------|-------|
| log_group_id   | String                                       | 日志组ID |
| log_group_name | String                                       | 日志组名称 |
| log_streams    | Array of <a href="#">log_streams</a> objects | 日志流   |

表 6-725 log\_streams

| 参数              | 参数类型   | 描述    |
|-----------------|--------|-------|
| log_stream_name | String | 日志流名称 |
| log_stream_id   | String | 日志流ID |

状态码：400

表 6-726 响应 Body 参数

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| error_code | String | 错误码<br>最小长度：8<br>最大长度：36   |
| error_msg  | String | 错误描述<br>最小长度：2<br>最大长度：512 |

## 请求示例

```
GET https://{endpoint}/v1/{project_id}/lts/{member_account_id}/all-streams
/v1/{project_id}/lts/{member_account_id}/all-streams
```

## 响应示例

状态码：200

组织成员日志组日志流

```
{
 "results": [{
 "log_group_name": "cuiss",
 "log_streams": [{
 "log_stream_name": "wjyTest",
 "log_stream_id": "d67124ed-b4bd-4d89-8646-5905c2eeaa62"
 }],
 "log_group_id": "d6c04fe9-170a-4d98-94e6-7844f47bd0b8"
 }]
}
```

## 状态码

| 状态码 | 描述             |
|-----|----------------|
| 200 | 组织成员日志组日志流     |
| 400 | Error response |

## 错误码

请参见[错误码](#)。

## 6.19.5 更新汇聚配置

### 功能介绍

只能由管理员或者委托管理员,更新汇聚配置

### 调用方法

请参见[如何调用API](#)。

### URI

PUT /v1/{project\_id}/lts/log-converge-config

表 6-727 路径参数

| 参数         | 是否必选 | 参数类型   | 描述                                                         |
|------------|------|--------|------------------------------------------------------------|
| project_id | 是    | String | 项目ID, 获取方式请参见: <a href="#">获取项目ID, 获取账号ID, 日志组ID、日志流ID</a> |

### 请求参数

表 6-728 请求 Header 参数

| 参数           | 是否必选 | 参数类型   | 描述                                                   |
|--------------|------|--------|------------------------------------------------------|
| X-Auth-Token | 是    | String | 从IAM服务获取的用户Token, 获取方式请参见: <a href="#">获取用户Token</a> |
| Content-Type | 是    | String | 该字段填为: application/json;charset=UTF-8。               |

表 6-729 请求 Body 参数

| 参数                | 是否必选 | 参数类型   | 描述       |
|-------------------|------|--------|----------|
| id                | 否    | String | ID       |
| member_account_id | 是    | String | 组织成员账号id |
| member_project_id | 否    | String | 组织成员项目id |

| 参数                    | 是否必选 | 参数类型                                              | 描述             |
|-----------------------|------|---------------------------------------------------|----------------|
| create_time           | 否    | Long                                              | 创建时间           |
| update_time           | 否    | Long                                              | 更新时间           |
| status                | 否    | String                                            | 状态             |
| organization_id       | 是    | String                                            | 组织id           |
| management_account_id | 是    | String                                            | 管理员或者委托管理员账号id |
| management_project_id | 是    | String                                            | 管理员或者委托管理员项目id |
| version               | 否    | String                                            | 版本             |
| log_mapping_config    | 否    | Array of <a href="#">LogMappingConfig</a> objects | 日志汇聚配置         |

表 6-730 LogMappingConfig

| 参数                    | 是否必选 | 参数类型                                                  | 描述      |
|-----------------------|------|-------------------------------------------------------|---------|
| source_log_group_id   | 是    | String                                                | 源日志组ID  |
| target_log_group_id   | 否    | String                                                | 目标日志组ID |
| target_log_group_name | 是    | String                                                | 目标日志组名称 |
| log_stream_config     | 否    | Array of <a href="#">LogMappingStreamInfo</a> objects | 日志流配置   |

表 6-731 LogMappingStreamInfo

| 参数                   | 是否必选 | 参数类型   | 描述      |
|----------------------|------|--------|---------|
| source_log_stream_id | 是    | String | 源日志流ID  |
| target_log_stream_id | 否    | String | 目标日志流ID |

| 参数                       | 是否必选 | 参数类型    | 描述                                         |
|--------------------------|------|---------|--------------------------------------------|
| target_log_stream_name   | 是    | String  | 目标日志流名称                                    |
| target_log_stream_eps_id | 否    | String  | 目标日志流EPS ID                                |
| target_log_stream_ttl    | 是    | Integer | 目标日志流ttl保存时间,单位时间(天).<br>最小值: 1<br>最大值: 30 |

## 响应参数

状态码: 200

表 6-732 响应 Body 参数

| 参数     | 参数类型   | 描述       |
|--------|--------|----------|
| status | String | 更新汇聚配置状态 |

## 请求示例

```
PUT https://{endpoint}/v1/{project_id}/lts/log-converge-config
{
 "management_account_id": "115ba1c5fec44839820795ebefe25f2a",
 "management_project_id": "a5baef478e6840dc9f454bbc00c996c6",
 "organization_id": "o-fa9h4ghf0hnpfnq4xd02tah2nxmtugrf",
 "member_account_id": "1d26cc8c86a840e28a4f8d0d07852f1d",
 "log_mapping_config": [{
 "source_log_group_id": "21845482-07fc-440c-a26a-840e6cb12289",
 "target_log_group_id": "",
 "target_log_group_name": "lts-xxx_xx-xxx-test",
 "log_stream_config": [{
 "source_log_stream_id": "72ece92e-af01-4216-aaa7-6df7a66465c7",
 "target_log_stream_eps_id": "0",
 "target_log_stream_id": "",
 "target_log_stream_name": "lts_xxx_xxx_xxx_xxx",
 "target_log_stream_ttl": 180
 }]
 }]
}
```

## 响应示例

状态码: 200

更新汇聚配置响应体

```
{
 "status": "creating"
}
```

## SDK 代码示例

SDK代码示例如下。

### Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.lts.v2.region.LtsRegion;
import com.huaweicloud.sdk.lts.v2.*;
import com.huaweicloud.sdk.lts.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateLogConvergeConfigSolution {

 public static void main(String[] args) {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
 // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
 // environment variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running
 // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 String ak = System.getenv("CLOUD_SDK_AK");
 String sk = System.getenv("CLOUD_SDK_SK");
 String projectId = "{project_id}";

 ICredential auth = new BasicCredentials()
 .withProjectId(projectId)
 .withAk(ak)
 .withSk(sk);

 LtsClient client = LtsClient.newBuilder()
 .withCredential(auth)
 .withRegion(LtsRegion.valueOf("<YOUR REGION>"))
 .build();
 UpdateLogConvergeConfigRequest request = new UpdateLogConvergeConfigRequest();
 UpdateLogConvergeConfig body = new UpdateLogConvergeConfig();
 List<LogMappingStreamInfo> listLogMappingConfigLogStreamConfig = new ArrayList<>();
 listLogMappingConfigLogStreamConfig.add(
 new LogMappingStreamInfo()
 .withSourceLogStreamId("72ece92e-af01-4216-aaa7-6df7a66465c7")
 .withTargetLogStreamId("")
 .withTargetLogStreamName("lts_xxx_xxx_xxx_xxx")
 .withTargetLogStreamEpsId("0")
 .withTargetLogStreamTtl(180)
);
 List<LogMappingConfig> listbodyLogMappingConfig = new ArrayList<>();
 listbodyLogMappingConfig.add(
 new LogMappingConfig()
 .withSourceLogGroupId("21845482-07fc-440c-a26a-840e6cb12289")
 .withTargetLogGroupId("")
 .withTargetLogGroupName("lts-xxx_xx-xxx-test")
 .withLogStreamConfig(listLogMappingConfigLogStreamConfig)
);
 body.withLogMappingConfig(listbodyLogMappingConfig);
 body.withManagementProjectId("a5baef478e6840dc9f454bbc00c996c6");
 body.withManagementAccountId("115ba1c5fec44839820795ebefe25f2a");
 body.withOrganizationId("o-fa9h4ghf0hnpfnq4xd02tah2nxmtugrf");
 body.withMemberAccountId("1d26cc8c86a840e28a4f8d0d07852f1d");
 request.withBody(body);
 try {
 UpdateLogConvergeConfigResponse response = client.updateLogConvergeConfig(request);
 System.out.println(response.toString());
 }
 }
}
```

```
} catch (ConnectionException e) {
 e.printStackTrace();
} catch (RequestTimeoutException e) {
 e.printStackTrace();
} catch (ServiceResponseException e) {
 e.printStackTrace();
 System.out.println(e.getHttpStatusCode());
 System.out.println(e.getRequestId());
 System.out.println(e.getErrorCode());
 System.out.println(e.getErrorMsg());
}
}
```

## Python

```
coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsklts.v2.region.lts_region import LtsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsklts.v2 import *

if __name__ == "__main__":
 # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 # variables and decrypted during use to ensure security.
 # In this example, AK and SK are stored in environment variables for authentication. Before running this
 # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak = os.environ["CLOUD_SDK_AK"]
 sk = os.environ["CLOUD_SDK_SK"]
 projectId = "{project_id}"

 credentials = BasicCredentials(ak, sk, projectId)

 client = LtsClient.new_builder() \
 .with_credentials(credentials) \
 .with_region(LtsRegion.value_of("<YOUR REGION>")) \
 .build()

 try:
 request = UpdateLogConvergeConfigRequest()
 listLogStreamConfigLogMappingConfig = [
 LogMappingStreamInfo(
 source_log_stream_id="72ece92e-af01-4216-aaa7-6df7a66465c7",
 target_log_stream_id="",
 target_log_stream_name="lts-xxx-xxx-xxx-xxx",
 target_log_stream_eps_id="0",
 target_log_stream_ttl=180
)
]
 listLogMappingConfigbody = [
 LogMappingConfig(
 source_log_group_id="21845482-07fc-440c-a26a-840e6cb12289",
 target_log_group_id="",
 target_log_group_name="lts-xxx-xx-xxx-test",
 log_stream_config=listLogStreamConfigLogMappingConfig
)
]
 request.body = UpdateLogConvergeConfig(
 log_mapping_config=listLogMappingConfigbody,
 management_project_id="a5baef478e6840dc9f454bbc00c996c6",
 management_account_id="115ba1c5fec44839820795ebefe25f2a",
 organization_id="o-fa9h4ghf0hnpfnq4xd02tah2nxmtugrf",
 member_account_id="1d26cc8c86a840e28a4f8d0d07852f1d"
)
 response = client.update_log_converge_config(request)
 print(response)
```

```
except exceptions.ClientRequestException as e:
 print(e.status_code)
 print(e.request_id)
 print(e.error_code)
 print(e.error_msg)
```

## Go

```
package main

import (
 "fmt"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
 lts "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2"
 "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/model"
 region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/lts/v2/region"
)

func main() {
 // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
 // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
 // variables and decrypted during use to ensure security.
 // In this example, AK and SK are stored in environment variables for authentication. Before running this
 // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
 ak := os.Getenv("CLOUD_SDK_AK")
 sk := os.Getenv("CLOUD_SDK_SK")
 projectId := "{project_id}"

 auth := basic.NewCredentialsBuilder().
 WithAk(ak).
 WithSk(sk).
 WithProjectId(projectId).
 Build()

 client := lts.NewLtsClient(
 lts.LtsClientBuilder().
 WithRegion(region.ValueOf("<YOUR REGION>")).
 WithCredential(auth).
 Build())

 request := &model.UpdateLogConvergeConfigRequest{
 targetLogStreamIdLogStreamConfig:= ""
 targetLogStreamEpsIdLogStreamConfig:= "0"
 var listLogStreamConfigLogMappingConfig = []model.LogMappingStreamInfo{
 {
 SourceLogStreamId: "72ece92e-af01-4216-aaa7-6df7a66465c7",
 TargetLogStreamId: &targetLogStreamIdLogStreamConfig,
 TargetLogStreamName: "lts-xxx-xxx-xxx-xxx",
 TargetLogStreamEpsId: &targetLogStreamEpsIdLogStreamConfig,
 TargetLogStreamTtl: int32(180),
 },
 }
 targetLogGroupIdLogMappingConfig:= ""
 var listLogMappingConfigbody = []model.LogMappingConfig{
 {
 SourceLogGroupId: "21845482-07fc-440c-a26a-840e6cb12289",
 TargetLogGroupId: &targetLogGroupIdLogMappingConfig,
 TargetLogGroupName: "lts-xxx-xx-xxx-test",
 LogStreamConfig: &listLogStreamConfigLogMappingConfig,
 },
 }
 request.Body = &model.UpdatedLogConvergeConfig{
 LogMappingConfig: &listLogMappingConfigbody,
 ManagementProjectId: "a5baef478e6840dc9f454bbc00c996c6",
 ManagementAccountId: "115ba1c5fec44839820795ebefe25f2a",
 OrganizationId: "o-fa9h4ghf0hnpfnq4xd02tah2nxmtugrf",
 MemberAccountId: "1d26cc8c86a840e28a4f8d0d07852f1d",
 }
 response, err := client.UpdateLogConvergeConfig(request)
```

```
if err == nil {
 fmt.Printf("%+v\n", response)
} else {
 fmt.Println(err)
}
}
```

## 更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

## 状态码

| 状态码 | 描述        |
|-----|-----------|
| 200 | 更新汇聚配置响应体 |

## 错误码

请参见[错误码](#)。

# 7 权限和授权项

如果您需要对您所拥有的LTS进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称IAM），如果账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用LTS的其它功能。

默认情况下，新建的IAM用户没有任何权限，您需要将其加入用户组，并给用户组授予策略或角色，才能使用户组中的用户获得对应的权限，这一过程称为授权。授权后，用户就可以基于被授予的权限对LTS进行操作。

权限根据授权的精细程度，分为**角色**和**策略**。角色以服务为粒度，是IAM最初提供了一种根据用户的工作职能定义权限的粗粒度授权机制。策略以API接口为粒度进行权限拆分，授权更加精细，可以精确到某个操作、资源和条件，能够满足企业对权限最小化的安全管控要求。

## 📖 说明

如果您要允许或是禁止某个接口的操作权限，请使用策略。

账号具备所有接口的调用权限，如果使用账号下的IAM用户发起API请求时，该IAM用户必须具备调用该接口所需的权限，否则，API请求将调用失败。每个接口所需要的权限，与各个接口所对应的授权项相对应，只有发起请求的用户被授予授权项所对应的策略，该用户才能成功调用该接口。例如，用户要调用接口来查询指标，那么这个IAM用户被授予的策略中必须包含允许“aom:metric:get”的授权项，该接口才能调用成功。

## 支持的授权项

策略包含系统策略和自定义策略，如果系统策略不满足授权要求，管理员可以创建自定义策略，并通过给用户组授予自定义策略来进行精细的访问控制。策略支持的操作与API相对应，授权项列表说明如下：

- 权限：自定义策略中授权项定义的内容即为权限。
- 对应API接口：自定义策略实际调用的API接口。
- 授权项：自定义策略中支持的Action，在自定义策略中的Action中写入授权项，可以实现授权项对应的权限功能。
- 依赖的授权项：部分Action存在对其他Action的依赖，需要将依赖的Action同时写入授权项，才能实现对应的权限功能。
- IAM项目(Project)/企业项目(Enterprise Project)：自定义策略的授权范围，包括IAM项目与企业项目。授权范围如果同时支持IAM项目和企业项目，表示此授权项

对应的自定义策略，可以在IAM和企业管理两个服务中给用户组授权并生效。如果仅支持IAM项目，不支持企业项目，表示仅能在IAM中给用户组授权并生效，如果在企业管理中授权，则该自定义策略不生效。关于IAM项目与企业项目的区别，详情请参见[IAM与企业项目的区别](#)。

**说明**

“√”表示支持，“x”表示暂不支持。

| 权限             | 对应API接口                                                               | 授权项 (Action)      | 依赖的授权项 | IA M项目 (Pr oject) | 企业项目 (Enter prise Projec t) |
|----------------|-----------------------------------------------------------------------|-------------------|--------|-------------------|-----------------------------|
| 创建日志组          | POST /v2/{project_id}/groups                                          | lts:groups:create | -      | √                 | x                           |
| 查询账号下所有日志组     | GET /v2/{project_id}/groups                                           | lts:groups:list   | -      | √                 | x                           |
| 修改日志组          | POST /v2/{project_id}/groups/{log_group_id}                           | lts:groups:put    | -      | √                 | x                           |
| 删除日志组          | DELETE /v2/{project_id}/groups/{log_group_id}                         | lts:groups:delete | -      | √                 | x                           |
| 创建日志流          | POST /v2/{project_id}/groups/{log_group_id}/streams                   | lts:topics:create | -      | √                 | √                           |
| 查询指定日志组下的所有日志流 | GET /v2/{project_id}/groups/{log_group_id}/streams                    | lts:topics:list   | -      | √                 | √                           |
| 删除日志流          | DELETE /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id} | lts:topics:delete | -      | √                 | √                           |

| 权限       | 对应API接口                                                                                  | 授权项 ( Action )        | 依赖的授权项                                                                                                                                                                                 | IA M 项目 (Pr oje ct) | 企业项目 (Enter prise Projec t) |
|----------|------------------------------------------------------------------------------------------|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|-----------------------------|
| 查询日志     | POST /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/content/query        | lts:logs:list         | -                                                                                                                                                                                      | √                   | x                           |
| 查询结构化日志  | POST /v2/{project_id}/groups/{log_group_id}/streams/{log_stream_id}/struct-content/query | lts:logs:list         | -                                                                                                                                                                                      | √                   | x                           |
| OBS转储    | POST /v2/{project_id}/log-dump/obs                                                       | lts:transfer:s:create | obs:bucket:CreateBucket<br>obs:bucket:HeadBucket<br>obs:bucket:GetLifecycleConfiguration<br>obs:bucket:PutLifecycleConfiguration<br>obs:bucket:GetBucketAcl<br>obs:bucket:PutBucketAcl | √                   | x                           |
| 关闭超额采集开关 | POST /v2/{project_id}/collection/disable                                                 | aom:quota:set         | -                                                                                                                                                                                      | √                   | x                           |
| 打开超额采集开关 | POST /v2/{project_id}/collection/enable                                                  | aom:quota:set         | -                                                                                                                                                                                      | √                   | x                           |

# 8 附录

## 8.1 状态码

状态码如[表1](#)所示

表 8-1 状态码

| 状态码 | 返回值                   | 状态码说明            |
|-----|-----------------------|------------------|
| 200 | OK                    | GET和PUT操作正常返回。   |
| 201 | OK                    | POST请求成功，返回查询结果。 |
| 204 | No Content            | DELETE操作正常返回。    |
| 400 | Bad Request           | 请求错误。            |
| 401 | Unauthorized          | 未提供认证信息，或认证信息错。  |
| 403 | Forbidden             | 请求页面被禁止访问。       |
| 404 | Not Found             | 服务器无法找到被请求的资源。   |
| 408 | Request Timeout       | 请求超出了服务器的等待时间。   |
| 429 | Too Many Requests     | 当前请求过多。          |
| 500 | Internal Server Error | 请求未完成，服务异常。      |
| 503 | Service Unavailable   | 系统暂时不可用，请求受限。    |

## 8.2 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

| 状态码 | 错误码      | 错误信息                                             | 描述             | 处理措施                      |
|-----|----------|--------------------------------------------------|----------------|---------------------------|
| 400 | LTS.0009 | Failed to validate the request body.             | 参数校验失败。        | 请根据请求返回的错误信息修改请求参数后重试。    |
| 400 | LTS.0010 | The system encountered an internal error         | 系统内部错误         | LTS服务内部异常，此时请联系技术支持工程师处理。 |
| 400 | LTS.0201 | The log group is not existed                     | 创建日志流失败，日志组不存在 | 核对groupId是否存在。            |
| 400 | LTS.0208 | Log stream is associated by transfer             | 日志流不存在         | 请确认要删除的日志流是否存在            |
| 400 | LTS.0301 | '*' and '?' not allowed as first character       | *和? 放在关键字中间或末尾 | 请根据错误信息检查Keywords字段。      |
| 400 | LTS.2001 | Failed to create alarm rule                      | 创建错误           | 核对projectId是否正常。          |
| 400 | LTS.2002 | Failed to delete alarm rule                      | 删除错误           | 查看数据库服务是否正常或网络连接是否正常。     |
| 400 | LTS.2003 | Failed to update alarm rule                      | 修改错误           | 连接数据库异常，检查数据库实例状态         |
| 400 | LTS.2004 | The size of alarm rule has exceed the limit: 200 | 告警规则条数不能大于200条 | 请删除原有告警规则。                |
| 400 | LTS.2005 | The parameter is incorrect                       | 参数有误           | 根据返回错误信息检查参数。             |
| 400 | LTS.2006 | Alarm rule name has already exist                | 告警规则名称存在       | 请检查告警规则名称是否存在。            |
| 400 | LTS.2007 | Alarm rule not exist                             | 告警规则不存在        | 请检查告警规则是否存在。              |
| 400 | LTS.2008 | Find Alarm rule failed                           | 告警规则查询失败       | 查看数据库服务是否正常或网络连接是否正常。     |

| 状态码 | 错误码      | 错误信息                                                                                                     | 描述                       | 处理措施                                 |
|-----|----------|----------------------------------------------------------------------------------------------------------|--------------------------|--------------------------------------|
| 400 | LTS.2009 | User must have SMN service authority                                                                     | 用户必须拥有SMN服务权限            | 查看用户是否拥有SMN服务权限。                     |
| 400 | LTS.2010 | Topics cannot be empty                                                                                   | SMN主题不能为空                | 请检查主题是否为空。                           |
| 400 | LTS.2011 | Alarm rule invalid query frequency or invalid cron expression                                            | 统计周期超限或cron表达式错误         | 请检查统计周期或cron表达式。                     |
| 400 | LTS.2012 | The query time range cannot be larger than 1 hour when the query frequency is less than every 5 minutes. | 统计周期小于5分钟时，查询时间范围不能大于1小时 | 请检查统计周期与查询范围。                        |
| 400 | LTS.2013 | Send Subject Error                                                                                       | 发送SMN主题错误                | 请根据错误信息检查主题。                         |
| 403 | LTS.0003 | No permission.                                                                                           | 非法的角色                    | 请授予LTS服务权限                           |
| 403 | LTS.0011 | Invalid projectId                                                                                        | 非法的projectId             | 请确认URL中的projectId与token中的projectId一致 |
| 500 | LTS.0107 | Current user does not have the permission to operate this group                                          | 更新日志组失败                  | 连接数据库异常,检查数据库实例状态                    |

## 8.3 获取账号 ID、项目 ID、日志组 ID、日志流 ID

### 获取账号 ID 和项目 ID

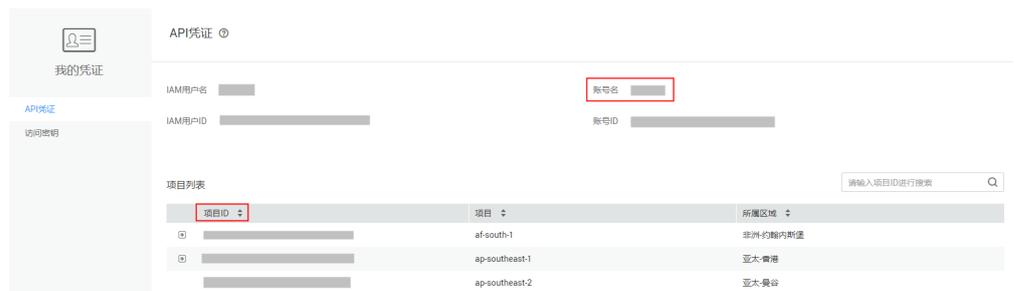
在调用接口的时候，部分URL中需要填入账号ID（domain-id）和项目ID，获取步骤如下：

1. 注册并登录管理控制台。

- 单击用户名，在下拉列表中单击“我的凭证”。



- 在“我的凭证”页面查看账号ID和项目ID。



多项目时，展开“所属区域”，从“项目ID”列获取子项目ID。

## 获取日志组和日志流 ID

- 登录云日志服务LTS管理控制台。
- 在“日志管理”页面，将鼠标悬浮在日志组名称上，即可查看日志组ID。
- 单击日志组名称对应的  按钮，将鼠标悬浮在日志流名称上，即可查看日志流ID。

## 8.4 OBS 转储时区表

| 名称       | Name                          | id           | timezone  |
|----------|-------------------------------|--------------|-----------|
| 国际日期变更线西 | International Date Line West  | Etc/GMT+12   | UTC-12:00 |
| 协调世界时-11 | Coordinated Universal Time-11 | Etc/GMT+11   | UTC-11:00 |
| 阿留申群岛    | Aleutian Islands              | America/Adak | UTC-10:00 |

| 名称              | Name                                | id                     | timezone  |
|-----------------|-------------------------------------|------------------------|-----------|
| 夏威夷             | Hawaii                              | Pacific/Honolulu       | UTC-10:00 |
| 马克萨斯群岛          | Marquesas Islands                   | Pacific/Marquesas      | UTC-09:30 |
| 阿拉斯加            | Alaska                              | America/Anchorage      | UTC-09:00 |
| 协调世界时-09        | Coordinated Universal Time-09       | Etc/GMT+9              | UTC-09:00 |
| 太平洋时间（美国和加拿大）   | -                                   | America/Los_Angeles    | UTC-08:00 |
| 下加利福尼亚州         | Baja California                     | America/Tijuana        | UTC-08:00 |
| 协调世界时-08        | Coordinated Universal Time-08       | Etc/GMT+8              | UTC-08:00 |
| 奇瓦瓦，拉巴斯，马萨特兰    | Chihuahua, La Paz, Mazatlan         | America/Chihuahua      | UTC-07:00 |
| 山地时间（美国和加拿大）    | -                                   | America/Denver         | UTC-07:00 |
| 亚利桑那            | Arizona                             | America/Phoenix        | UTC-07:00 |
| 中部时间（美国和加拿大）    | -                                   | America/Chicago        | UTC-06:00 |
| 中美洲             | Central America                     | America/Guatemala      | UTC-06:00 |
| 瓜达拉哈拉，墨西哥城，蒙特雷  | Guadalajara, Mexico City, Monterrey | America/Mexico_City    | UTC-06:00 |
| 萨斯喀彻温           | Saskatchewan                        | America/Regina         | UTC-06:00 |
| 复活节岛            | Easter Island                       | Pacific/Easter         | UTC-06:00 |
| 波哥大，利马，基多，里奥布朗库 | Bogota, Lima, Quito, Rio Branco     | America/Bogota         | UTC-05:00 |
| 切图马尔            | Chetumal                            | America/Cancun         | UTC-05:00 |
| 印第安纳州（东部）       | -                                   | America/Indianapolis   | UTC-05:00 |
| 秘鲁              | Peru                                | America/Lima           | UTC-05:00 |
| 东部时间（美国和加拿大）    | -                                   | America/New_York       | UTC-05:00 |
| 海地              | Haiti                               | America/Port-au-Prince | UTC-05:00 |
| 亚松森             | Asuncion                            | America/Asuncion       | UTC-04:00 |
| 加拉加斯            | Caracas                             | America/Caracas        | UTC-04:00 |
| 库亚巴             | Cuiaba                              | America/Cuiaba         | UTC-04:00 |

| 名称              | Name                                 | id                             | timezone   |
|-----------------|--------------------------------------|--------------------------------|------------|
| 特克斯和凯科斯群岛       | Turks and Caicos                     | America/Grand_Turk             | UTC-04:00  |
| 大西洋时间（加拿大）      | -                                    | America/Halifax                | UTC-04:00  |
| 乔治敦，拉巴斯，玛瑙斯，圣胡安 | Georgetown, La Paz, Manaus, San Juan | America/La_Paz                 | UTC-04:00  |
| 圣地亚哥            | Santiago                             | America/Santiago               | UTC-04:00  |
| 纽芬兰             | Newfoundland                         | America/St_Johns               | UTC-03:30  |
| 阿拉瓜伊纳           | Araguaina                            | America/Araguaina              | UTC-03:00  |
| 阿根廷             | Argentina                            | America/Argentina/Buenos_Aires | UTC-03:00  |
| 萨尔瓦多            | Salvador                             | America/Bahia                  | UTC-03:00  |
| 布宜诺斯艾利斯         | City of Buenos Aires                 | America/Buenos_Aires           | UTC-03:00  |
| 卡宴，福塔雷萨         | Cayenne, Fortaleza                   | America/Cayenne                | UTC-03:00  |
| 格陵兰             | Greenland                            | America/Godthab                | UTC-03:00  |
| 圣皮埃尔和密克隆群岛      | Saint Pierre and Miquelon            | America/Miquelon               | UTC-03:00  |
| 蒙得维的亚           | Montevideo                           | America/Montevideo             | UTC-03:00  |
| 巴西利亚            | Brasilia                             | America/Sao_Paulo              | UTC-03:00  |
| 协调世界时-02        | Coordinated Universal Time-02        | Etc/GMT+2                      | UTC-02:00  |
| 亚速尔群岛           | Azores                               | Atlantic/Azores                | UTC-01:00  |
| 佛得角群岛           | Cabo Verde Is.                       | Atlantic/Cape_Verde            | UTC-01:00  |
| 加纳              | Ghana                                | Africa/Accra                   | UTC        |
| 协调世界时间          | Coordinated Universal Time           | Etc/GMT                        | UTC        |
| 蒙罗维亚，雷克雅未克      | Monrovia, Reykjavik                  | Atlantic/Reykjavik             | UTC +00:00 |
| 卡萨布兰卡           | Casablanca                           | Africa/Casablanca              | UTC +00:00 |
| 都柏林，爱丁堡，里斯本，伦敦  | Dublin, Edinburgh, Lisbon, London    | Europe/London                  | UTC +00:00 |
| 中非西部            | West Central Africa                  | Africa/Lagos                   | UTC +01:00 |

| 名称                                     | Name                                                    | id                  | timezone      |
|----------------------------------------|---------------------------------------------------------|---------------------|---------------|
| 安哥拉                                    | Angola                                                  | Africa/Luanda       | UTC<br>+01:00 |
| 温得和克                                   | Windhoek                                                | Africa/Windhoek     | UTC<br>+01:00 |
| 阿姆斯特丹, 柏林,<br>伯尔尼, 罗马, 斯德<br>哥尔摩, 维也纳  | Amsterdam, Berlin,<br>Bern, Rome,<br>Stockholm, Vienna  | Europe/Berlin       | UTC<br>+01:00 |
| 贝尔格莱德, 布拉迪<br>斯拉发, 布达佩斯,<br>卢布尔雅那, 布拉格 | Belgrade, Bratislava,<br>Budapest, Ljubljana,<br>Prague | Europe/Budapest     | UTC<br>+01:00 |
| 布鲁塞尔, 哥本哈<br>根, 马德里, 巴黎                | Brussels,<br>Copenhagen,<br>Madrid, Paris               | Europe/Paris        | UTC<br>+01:00 |
| 萨拉热窝, 斯科普<br>里, 华沙, 萨格勒布               | Sarajevo, Skopje,<br>Warsaw, Zagreb                     | Europe/Warsaw       | UTC<br>+01:00 |
| 开罗                                     | Cairo                                                   | Africa/Cairo        | UTC<br>+02:00 |
| 博茨瓦纳                                   | Botswana                                                | Africa/Gaborone     | UTC<br>+02:00 |
| 津巴布韦                                   | Zimbabwe                                                | Africa/Harare       | UTC<br>+02:00 |
| 哈拉雷, 比勒陀利亚                             | Harare, Pretoria                                        | Africa/Johannesburg | UTC<br>+02:00 |
| 赞比亚                                    | Zambia                                                  | Africa/Lusaka       | UTC<br>+02:00 |
| 莫桑比克                                   | Mozambique                                              | Africa/Maputo       | UTC<br>+02:00 |
| 的黎波里                                   | Tripoli                                                 | Africa/Tripoli      | UTC<br>+02:00 |
| 安曼                                     | Amman                                                   | Asia/Amman          | UTC<br>+02:00 |
| 贝鲁特                                    | Beirut                                                  | Asia/Beirut         | UTC<br>+02:00 |
| 加沙、希伯伦                                 | Gaza, Hebron                                            | Asia/Hebron         | UTC<br>+02:00 |
| 耶路撒冷                                   | Jerusalem                                               | Asia/Jerusalem      | UTC<br>+02:00 |
| 雅典, 布加勒斯特                              | Athens, Bucharest                                       | Europe/Bucharest    | UTC<br>+02:00 |

| 名称                          | Name                                          | id                   | timezone      |
|-----------------------------|-----------------------------------------------|----------------------|---------------|
| 基希讷乌                        | Chisinau                                      | Europe/Chisinau      | UTC<br>+02:00 |
| 加里宁格勒                       | Kaliningrad                                   | Europe/Kaliningrad   | UTC<br>+02:00 |
| 赫尔辛基, 基辅, 里加, 索非亚, 塔林, 维尔纽斯 | Helsinki, Kyiv, Riga, Sofia, Tallinn, Vilnius | Europe/Kiev          | UTC<br>+02:00 |
| 埃塞俄比亚                       | Ethiopia                                      | Africa/Addis_Ababa   | UTC<br>+03:00 |
| 坦桑尼亚                        | Tanzania                                      | Africa/Dar_es_Salaam | UTC<br>+03:00 |
| 内罗毕                         | Nairobi                                       | Africa/Nairobi       | UTC<br>+03:00 |
| 巴格达                         | Baghdad                                       | Asia/Baghdad         | UTC<br>+03:00 |
| 科威特, 利雅得                    | Kuwait, Riyadh                                | Asia/Riyadh          | UTC<br>+03:00 |
| 伊斯坦布尔                       | Istanbul                                      | Europe/Istanbul      | UTC<br>+03:00 |
| 明斯克                         | Minsk                                         | Europe/Minsk         | UTC<br>+03:00 |
| 莫斯科, 圣彼得堡, 伏尔加格勒            | Moscow, St. Petersburg, Volgograd             | Europe/Moscow        | UTC<br>+03:00 |
| 巴库                          | Baku                                          | Asia/Baku            | UTC<br>+04:00 |
| 阿布扎比, 马斯喀特                  | Abu Dhabi, Muscat                             | Asia/Dubai           | UTC<br>+04:00 |
| 第比利斯                        | Tbilisi                                       | Asia/Tbilisi         | UTC<br>+04:00 |
| 埃里温                         | Yerevan                                       | Asia/Yerevan         | UTC<br>+04:00 |
| 阿斯特拉罕, 乌拉扬诺夫斯克              | Astrakhan, Ulyanovsk                          | Europe/Astrakhan     | UTC<br>+04:00 |
| 伊热夫斯克, 萨马拉                  | Izhevsk, Samara                               | Europe/Samara        | UTC<br>+04:00 |
| 路易港                         | Port Louis                                    | Indian/Mauritius     | UTC<br>+04:00 |

| 名称                | Name                                | id                 | timezone      |
|-------------------|-------------------------------------|--------------------|---------------|
| 喀布尔               | Kabul                               | Asia/Kabul         | UTC<br>+04:30 |
| 伊斯兰堡, 卡拉奇         | Islamabad, Karachi                  | Asia/Karachi       | UTC<br>+05:00 |
| 阿什哈巴德, 塔什干        | Ashgabat, Tashkent                  | Asia/Tashkent      | UTC<br>+05:00 |
| 叶卡捷琳堡             | Ekaterinburg                        | Asia/Yekaterinburg | UTC<br>+05:00 |
| 钦奈, 加尔各答, 孟买, 新德里 | Chennai, Kolkata, Mumbai, New Delhi | Asia/Calcutta      | UTC<br>+05:30 |
| 斯里加亚渥登普拉          | Sri Jayawardenepura                 | Asia/Colombo       | UTC<br>+05:30 |
| 印度                | India                               | Asia/Kolkata       | UTC<br>+05:30 |
| 尼泊尔               | Nepal                               | Asia/Kathmandu     | UTC<br>+05:45 |
| 加德满都              | Kathmandu                           | Asia/Katmandu      | UTC<br>+05:45 |
| 阿斯塔纳              | Astana                              | Asia/Almaty        | UTC<br>+06:00 |
| 达卡                | Dhaka                               | Asia/Dhaka         | UTC<br>+06:00 |
| 鄂木斯克              | Omsk                                | Asia/Omsk          | UTC<br>+06:00 |
| 仰光                | -                                   | Asia/Rangoon       | UTC<br>+06:30 |
| 缅甸                | Myanmar                             | Asia/Yangon        | UTC<br>+06:30 |
| 曼谷, 河内, 雅加达       | Bangkok, Hanoi, Jakarta             | Asia/Bangkok       | UTC<br>+07:00 |
| 巴尔瑙尔, 戈尔诺-阿尔泰斯克   | Barnaul, Gorno-Altaysk              | Asia/Barnaul       | UTC<br>+07:00 |
| 越南                | Vietnam                             | Asia/Ho_Chi_Minh   | UTC<br>+07:00 |
| 科布多               | Hovd                                | Asia/Hovd          | UTC<br>+07:00 |
| 印度尼西亚             | Indonesia                           | Asia/Jakarta       | UTC<br>+07:00 |

| 名称                    | Name                                  | id                | timezone      |
|-----------------------|---------------------------------------|-------------------|---------------|
| 克拉斯诺亚尔斯克              | Krasnoyarsk                           | Asia/Krasnoyarsk  | UTC<br>+07:00 |
| 新西伯利亚                 | Novosibirsk                           | Asia/Novosibirsk  | UTC<br>+07:00 |
| 柬埔寨                   | Cambodia                              | Asia/Phnom_Penh   | UTC<br>+07:00 |
| 托木斯克                  | Tomsk                                 | Asia/Tomsk        | UTC<br>+07:00 |
| 老挝                    | Laos                                  | Asia/Vientiane    | UTC<br>+07:00 |
| 中国香港                  | Hong Kong (China)                     | Asia/Hong_Kong    | UTC<br>+08:00 |
| 伊尔库茨克                 | Irkutsk                               | Asia/Irkutsk      | UTC<br>+08:00 |
| 马来西亚                  | Malaysia                              | Asia/Kuala_Lumpur | UTC<br>+08:00 |
| 中国澳门                  | Macau (China)                         | Asia/Macau        | UTC<br>+08:00 |
| 菲律宾                   | Philippines                           | Asia/Manila       | UTC<br>+08:00 |
| 北京, 重庆, 香港特别行政区, 乌鲁木齐 | Beijing, Chongqing, Hong Kong, Urumqi | Asia/Shanghai     | UTC<br>+08:00 |
| 吉隆坡, 新加坡              | Kuala Lumpur, Singapore               | Asia/Singapore    | UTC<br>+08:00 |
| 台北                    | Taipei                                | Asia/Taipei       | UTC<br>+08:00 |
| 乌兰巴托                  | Ulaanbaatar                           | Asia/Ulaanbaatar  | UTC<br>+08:00 |
| 珀斯                    | Perth                                 | Australia/Perth   | UTC<br>+08:00 |
| 尤克拉                   | Eucla                                 | Australia/Eucla   | UTC<br>+08:45 |
| 赤塔市                   | Chita                                 | Asia/Chita        | UTC<br>+09:00 |
| 首尔                    | Seoul                                 | Asia/Seoul        | UTC<br>+09:00 |
| 大阪, 札幌, 东京            | Osaka, Sapporo, Tokyo                 | Asia/Tokyo        | UTC<br>+09:00 |

| 名称                 | Name                                    | id                   | timezone      |
|--------------------|-----------------------------------------|----------------------|---------------|
| 雅库茨克               | Yakutsk                                 | Asia/Yakutsk         | UTC<br>+09:00 |
| 阿德莱德               | Adelaide                                | Australia/Adelaide   | UTC<br>+09:30 |
| 达尔文                | Darwin                                  | Australia/Darwin     | UTC<br>+09:30 |
| 符拉迪沃斯托克            | Vladivostok                             | Asia/Vladivostok     | UTC<br>+10:00 |
| 布里斯班               | Brisbane                                | Australia/Brisbane   | UTC<br>+10:00 |
| 霍巴特                | Hobart                                  | Australia/Hobart     | UTC<br>+10:00 |
| 堪培拉, 墨尔本, 悉尼       | Canberra,<br>Melbourne, Sydney          | Australia/Sydney     | UTC<br>+10:00 |
| 关岛, 莫尔兹比港          | Guam, Port Moresby                      | Pacific/Port_Moresby | UTC<br>+10:00 |
| 豪勋爵岛               | Lord Howe Island                        | Australia/Lord_Howe  | UTC<br>+10:30 |
| 马加丹                | Magadan                                 | Asia/Magadan         | UTC<br>+11:00 |
| 萨哈林                | Sakhalin                                | Asia/Sakhalin        | UTC<br>+11:00 |
| 乔库尔达赫              | Chokurdakh                              | Asia/Srednekolymsk   | UTC<br>+11:00 |
| 布干维尔岛              | Bougainville Island                     | Pacific/Bougainville | UTC<br>+11:00 |
| 所罗门群岛, 新喀里多尼亚      | Solomon Is., New<br>Caledonia           | Pacific/Guadalcanal  | UTC<br>+11:00 |
| 诺福克岛               | Norfolk Island                          | Pacific/Norfolk      | UTC<br>+11:00 |
| 阿纳德尔, 堪察加彼得罗巴甫洛夫斯克 | Anadyr,<br>Petropavlovsk-<br>Kamchatsky | Asia/Kamchatka       | UTC<br>+12:00 |
| 协调世界时+12           | Coordinated<br>Universal Time+12        | Etc/GMT-12           | UTC<br>+12:00 |
| 奥克兰, 惠灵顿           | Auckland,<br>Wellington                 | Pacific/Auckland     | UTC<br>+12:00 |

| 名称    | Name              | id                 | timezone      |
|-------|-------------------|--------------------|---------------|
| 斐济    | Fiji              | Pacific/Fiji       | UTC<br>+12:00 |
| 查塔姆群岛 | Chatham Islands   | Pacific/Chatham    | UTC<br>+12:45 |
| 萨摩亚群岛 | Samoa             | Pacific/Apia       | UTC<br>+13:00 |
| 努库阿洛法 | Nuku alofa        | Pacific/Tongatapu  | UTC<br>+13:00 |
| 圣诞岛   | Kiritimati Island | Pacific/Kiritimati | UTC<br>+14:00 |