

云日志服务

API 参考

文档版本 01
发布日期 2022-08-15



版权所有 © 华为技术有限公司 2022。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。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <https://www.huawei.com>

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目录

1 使用前必读	1
1.1 概述	1
1.2 调用说明	1
1.3 终端节点	1
1.4 约束与限制	1
1.5 基本概念	1
2 API 概览	3
3 如何调用 API	4
3.1 构造请求	4
3.2 认证鉴权	7
3.3 返回结果	8
4 接口调用示例	10
5 API 说明	11
5.1 日志组管理	11
5.1.1 创建日志组	11
5.1.2 查询帐号下所有日志组	15
5.1.3 删除日志组	19
5.2 日志流管理	23
5.2.1 创建日志流	23
5.2.2 查询指定日志组下的所有日志流	27
5.2.3 删除日志流	31
5.3 日志管理	35
5.3.1 查询日志	35
5.4 日志转储	48
5.4.1 创建日志转储	48
6 附录	54
6.1 状态码	54
6.2 错误码	55
6.3 获取帐号 ID、项目 ID、日志组 ID、日志流 ID	57

1 使用前必读

1.1 概述

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。

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，删除日志组和日志流。

1.2 调用说明

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”。

1.3 终端节点

终端节点（Endpoint）即调用API的**请求地址**，不同服务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。

1.4 约束与限制

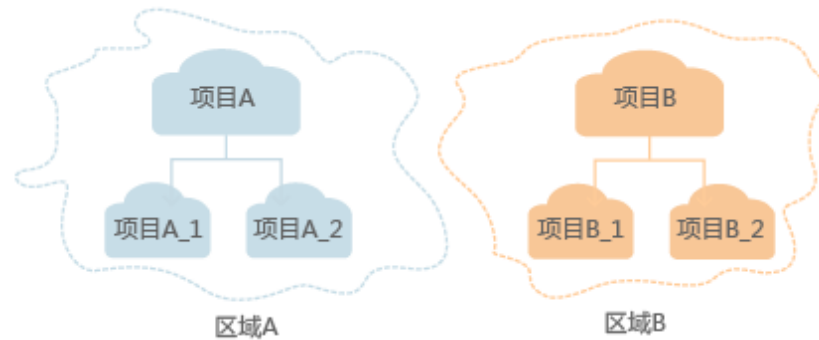
- 您能创建的LTS资源的数量与配额有关系。
- 更详细的限制请参见具体API的说明。

1.5 基本概念

- 用户
由帐号在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）。
通常在调用API的鉴权过程中，您需要用到帐号、用户和密码等信息。

- 区域（Region）
指云资源所在的物理位置，同一区域内可用区间内网互通，不同区域间内网不互通。通过在不同地区创建云资源，可以将应用程序设计的更接近特定客户的要求，或满足不同地区的法律或其他要求。
- 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，用户可以访问您帐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



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
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。

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[获取用户Token](#)说明如何调用API，该API获取用户的Token，Token可以用于调用其他API时鉴权。

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“亚太-吉隆坡”区域的Endpoint为“iam.my-kualalumpur-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“获取用户Token”API的resource-path为“/v3/auth/tokens”。
- **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。

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.my-kualalumpur-1.cloudalphi4.tnone.com.my/v3/auth/tokens
```

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
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 获取帐号ID、项目ID、日志组ID、日志流ID 章节获取项目编号。	否	e9993fc787d94b6c886cb aa340f9c0f4

名称	描述	是否必选	示例
X-Auth-Token	<p>用户Token。</p> <p>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。</p> <p>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。</p>	否 使用Token认证时该字段必选。	<p>注：以下仅为Token示例片段</p> <p>MIIPAgYJKoZlhvcNAQcCo...ggg1BBIIlNPXsidG9rZ</p>

📖 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[获取用户Token](#)接口，由于不需要认证，所以只添加“Content-Type”即可，添加消息头后的请求如下所示。

```
POST https://iam.my-kualalumpur-1.cloudalpai4.tnone.com.my/v3/auth/tokens
Content-Type: application/json
```

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。

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[获取用户Token](#)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，其中***username***为用户名，***domainname***为用户所属的帐号名称，***********为用户登录密码，***xxxxxxxxxxxxxxxxxxxx***为project的名称，您可以从[地区和终端节点](#)获取。

📖 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的作用域为某个帐号下所有资源或帐号的某个project下的资源，详细定义请参见[获取用户Token](#)。

```
POST https://iam.my-kualalumpur-1.cloudalpai4.tnone.com.my/v3/auth/tokens
Content-Type: application/json
```

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],

```

```
    "password": {
      "user": {
        "name": "username",
        "password": "*****#",
        "domain": {
          "name": "domainname"
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

到这里为止这个请求需要的内容就具备齐全了，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。

- Token认证：通过Token认证调用请求。
- 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

📖 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。

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": "xxxxxxx"  
}  
}
```

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.my-kualalumpur-1.cloudalpai4.tnone.com.my/v3/auth/tokens  
Content-Type: application/json  
X-Auth-Token: ABCDEFJ....
```

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。

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。

3.3 返回结果

状态码

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[错误码](#)。

对于“获取用户Token”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“获取用户Token”接口，返回如[图1](#)所示的消息头，其中“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。

图 3-1 获取用户 Token 响应消息头

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token → MIIVXQYJKoZIhvcNAQcCoIIYTCCEGoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOansiZXhwaXJlc19hdCI6IiwMTktMDItMTNUMC
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkajACgkqlqO1wi4JIGzrpd18LGXK5bdfq4lqHCYb8P4NaY0NYejcAgz/VeFYtLWT1GSO0zxKZmlQHq82HBqHdgIZO9fuEbL5dMhdavj+33wEI
xHRCE9I87o+k9-
j+CMZSE87bUGd5Uj6eRASXl1jipPEGA270g1FruooL6jggIFkNPQuFSOU8+uSsttVwrNfsC+qTp22Rkd5MCqFGQ8LcuUx3a+9CMBnOintWW7oeRUUVhVpxk8pxiX1wTEboX-
RzT6MUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==
x-xss-protection → 1; mode=block;
```

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“获取用户Token”接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "xxxxx",
            .....
          }
        ]
      }
    ]
  }
}
```

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": "The format of message is error",
  "error_code": "AS.0001"
}
```

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 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。

5.1 日志组管理

5.1.1 创建日志组

功能介绍

该功能指导用户创建日志管理的基本单位，用于日志存储及查询。目前每个帐户最多可以创建100个日志组。

URI

POST /v2/{project_id}/groups

表 5-1 路径参数

名称	是否必选	参数类型	说明
project_id	是	String	项目ID，获取方式请参见： 获取帐号ID、项目ID、日志组ID、日志流ID 。 缺省值：None 最小长度：32 最大长度：32

请求参数

表 5-2 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值: None 最小长度: 1000 最大长度: 2000
Content-Type	是	String	该字段填为: application/ json;charset=utf8。 缺省值: None 最小长度: 29 最大长度: 29

表 5-3 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。 最小长度: 1 最大长度: 64 枚举值: <ul style="list-style-type: none">• lts-group-01nh
ttl_in_days	是	Integer	日志存储时间(天)。 最大值: 7 枚举值: <ul style="list-style-type: none">• 7

响应参数

状态码为 201 时:

表 5-4 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

状态码为 400 时:**表 5-5 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

状态码为 401 时:**表 5-6 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

状态码为 403 时:**表 5-7 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

状态码为 500 时:

表 5-8 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

状态码为 503 时:

表 5-9 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

```
POST https://{endpoint}/v2/{project_id}/groups
/v2/{project_id}/groups
{
  "log_group_name": "lts-group-01nh",
  "ttl_in_days": 7
}
```

响应示例

状态码为 201 时:

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

状态码为 400 时:

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"
}
```

状态码为 401 时:

AuthFailed。鉴权失败，请确认token后再次请求。

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

状态码为 403 时:

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"
}
```

状态码为 500 时:

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"
}
```

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](#)。

5.1.2 查询帐号下所有日志组

功能介绍

该接口用于查询帐号下所有日志组。

URI

GET /v2/{project_id}/groups

表 5-10 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： 获取帐号ID、项目ID、日志组ID、日志流ID 。缺省值：None 最小长度：32 最大长度：32

请求参数

表 5-11 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值：None 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=utf8。 缺省值：None 最小长度：29 最大长度：29

响应参数

状态码为 200 时:

表 5-12 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 5-13 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：7
tag	Map<String,String>	日志组标签。

状态码为 401 时:

表 5-14 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

状态码为 403 时:

表 5-15 响应 Body 参数

参数	参数类型	描述
error_code	String	错误码。 枚举值： • LTS.0403

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

状态码为 500 时:

表 5-16 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

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

响应示例

状态码为 200 时:

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

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

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

状态码为 403 时:

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"  
}
```

状态码为 500 时:

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。
503	ServiceUnavailable。被请求的服务无效，服务不可用。

错误码

请参见[错误码](#)。

5.1.3 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。

URI

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

表 5-17 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： 获取帐号ID、项目ID、日志组ID、日志流ID 。缺省值：None 最小长度：32 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： 获取帐号ID、项目ID、日志组ID、日志流ID 。 缺省值：None 最小长度：36 最大长度：36

请求参数

表 5-18 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值：None 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=utf8。 缺省值：None 最小长度：29 最大长度：29

响应参数

状态码为 400 时:

表 5-19 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

状态码为 401 时:

表 5-20 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

状态码为 403 时:

表 5-21 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

状态码为 500 时:

表 5-22 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

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


响应示例

状态码为 400 时:

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"
}
```

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

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

状态码为 403 时:

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"
}
```

状态码为 500 时:

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。
503	ServiceUnavailable。被请求的服务无效，服务不可用。

错误码

请参见[错误码](#)。

5.2 日志流管理

5.2.1 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。

URI

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

表 5-23 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： 获取帐号ID、项目ID、日志组ID、日志流ID 。缺省值：None 最小长度：32 最大长度：32
log_group_id	是	String	租户想创建的日志流所在的日志组的groupid，一般为36位字符串。 缺省值：None 最小长度：36 最大长度：36

请求参数

表 5-24 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值：None 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=utf8。 缺省值: None 最小长度: 29 最大长度: 29

表 5-25 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 枚举值: • lts-stream-13ci
enterprise_project_name	否	String	企业项目名称。 最小长度: 1 最大长度: 64
ttl_in_days	否	Integer	日志存储时间(天)。 最小值: 1 最大值: 365 枚举值: 7

响应参数

状态码为 201 时:

表 5-26 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

状态码为 400 时:

表 5-27 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

状态码为 401 时:

表 5-28 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

状态码为 403 时:

表 5-29 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

状态码为 500 时:

表 5-30 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

状态码为 503 时:

表 5-31 响应 Body 参数

参数	参数类型	描述
-	String	ServiceUnavailable。被请求的服务无效, 服务不可用。

请求示例

```
POST https://{endpoint}/v2/{project_id}/groups/{log_group_id}/streams
/v2/{project_id}/groups/{log_group_id}/streams
{
  "log_stream_name": "lts-stream-02kh",
  "ttl_in_days": 7,
  "enterprise_project_name": "h2"
}
```

响应示例

状态码为 201 时:

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

状态码为 400 时:

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"
}
```

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

```
{
  "error_code": "LTS.0003",
```

```
"error_msg": "Invalid token"  
}
```

状态码为 403 时:

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"  
}
```

状态码为 500 时:

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"  
}
```

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。
503	ServiceUnavailable。被请求的服务无效，服务不可用。

错误码

请参见[错误码](#)。

5.2.2 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。

URI

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

表 5-32 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： 获取帐号ID、项目ID、日志组ID、日志流ID 。缺省值：None 最小长度：32 最大长度：32
log_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。 缺省值：None 最小长度：36 最大长度：36

请求参数

表 5-33 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值：None 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=utf8。 缺省值：None 最小长度：29 最大长度：29

响应参数

状态码为 200 时:

表 5-34 响应 Body 参数

参数	参数类型	描述
log_streams	Array of LogStream objects	日志流数组。

表 5-35 LogStream

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_name	String	日志流名称。 最小长度：36 最大长度：36
log_stream_id	String	日志流ID。 最小长度：36 最大长度：36
filter_count	Integer	过滤器个数。 最小值：0 最大值：5
tag	Map<String,String>	日志流标签。

状态码为 401 时:

表 5-36 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

状态码为 403 时:

表 5-37 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

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

状态码为 500 时:

表 5-38 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

状态码为 503 时:

表 5-39 响应 Body 参数

参数	参数类型	描述
-	String	ServiceUnavailable。被请求的服务无效, 服务不可用。

请求示例

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

响应示例

状态码为 200 时:

```
{
  "log_streams": [ {
    "creation_time":1630549842955,
    "log_stream_name":"lts-stream-02kh",
    "log_stream_id":"c54dbc58-0fd8-48ed-b007-6d54981427a7"
    "filter_count":0,
  } ]
}
```

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

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

状态码为 403 时:

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"
}
```

状态码为 500 时:

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。
503	ServiceUnavailable。被请求的服务无效, 服务不可用。

错误码

请参见[错误码](#)。

5.2.3 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。

URI

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

表 5-40 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	ID, 获取方式请参见: 获取帐号ID、项目ID、日志组ID、日志流ID 。 缺省值: None 最小长度: 32 最大长度: 32
log_group_id	是	String	租户想删除的日志流所在的日志组的groupid, 一般为36位字符串。 缺省值: None 最小长度: 36 最大长度: 36
log_stream_id	是	String	需要删除的日志流ID, 获取方式请参见: 获取帐号ID、项目ID、日志组ID、日志流ID 。 缺省值: None 最小长度: 36 最大长度: 36

请求参数

表 5-41 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值: None 最小长度: 1000 最大长度: 2000
Content-Type	是	String	该字段填为: application/json;charset=utf8。 缺省值: None 最小长度: 29 最大长度: 29

响应参数

状态码为 400 时:

表 5-42 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

状态码为 401 时:

表 5-43 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

状态码为 403 时:

表 5-44 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

状态码为 500 时:

表 5-45 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

状态码为 503 时:

表 5-46 响应 Body 参数

参数	参数类型	描述
-	String	ServiceUnavailable。被请求的服务无效, 服务不可用。

请求示例

```
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

状态码为 400 时:

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"
}
```

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

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

状态码为 403 时:

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"  
}
```

状态码为 500 时:

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。
503	ServiceUnavailable。被请求的服务无效，服务不可用。

错误码

请参见[错误码](#)。

5.3 日志管理

5.3.1 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。

URI

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

表 5-47 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	ID, 获取方式请参见: 获取帐号ID、项目ID、日志组ID、日志流ID 。 缺省值: None 最小长度: 32 最大长度: 32
log_group_id	是	String	日志组ID。 缺省值: None 最小长度: 36 最大长度: 36
log_stream_id	是	String	日志流ID。 缺省值: None 最小长度: 36 最大长度: 36

请求参数

表 5-48 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值: None 最小长度: 1000 最大长度: 2000
Content-Type	是	String	该字段填为: application/json;charset=utf8。 缺省值: None 最小长度: 29 最大长度: 29

表 5-49 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30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30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。 枚举值： <ul style="list-style-type: none">• 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。 最小长度：19 最大长度：19
is_desc	否	Boolean	顺序或者倒序查询，默认为false(顺序查询) 枚举值： <ul style="list-style-type: none">• 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： <ul style="list-style-type: none">• 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
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。 枚举值： true

响应参数

状态码为 200 时:

表 5-50 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。

表 5-51 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
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。

状态码为 400 时:

表 5-52 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

状态码为 401 时:

表 5-53 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

状态码为 403 时:

表 5-54 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

状态码为 500 时:

表 5-55 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

日志内容:

```
2020-07-25/14:44:42 this log is Error NO 1  
2020-07-25/14:44:43 this log is Error NO 2
```

```
2020-07-25/14:44:44 this log is Error NO 3
2020-07-25/14:44:45 this log is Error NO 4
2020-07-25/14:44:46 this log is Error NO 5
2020-07-25/14:44:47 this log is Error NO 6
2020-07-25/14:44:48 this log is Error NO 7
2020-07-25/14:44:49 this log is Error NO 8
2020-07-25/14:44:50 this log is Error NO 9
2020-07-25/14:44:51 this log is Error NO 10
```

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

- 首次查询日志

```
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”

```
{
  "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  
}
```

响应示例

- 首次查询日志

```
{  
  "count": 32,  
  "logs": [  
    {  
      "content": "2020-07-25/14:44:42 this <HighLightTag>log</HighLightTag> is Error NO 1\n",  
      "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\n",  
      "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\n",  
      "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\n",
  "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\n",
  "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\n",
  "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\n",
  "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\n",
    "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\n",
    "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\n",
    "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\n",
            "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\n",
  "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\n",
  "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\n",
      "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\n",
      "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\n",
  "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\n",
      "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\n",
      "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\n",
      "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\n",
      "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\n",
      "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\n",
      "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"
    }
  ]
}
```

```
    ]  
  }
```

状态码为 400 时:

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"  
}
```

状态码为 401 时:

AuthFailed。鉴权失败, 请确认token后再次请求。

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

状态码为 403 时:

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"  
}
```

状态码为 500 时:

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"  
}
```

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。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](#)。

5.4 日志转储

5.4.1 创建日志转储

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。

URI

POST /v2/{project_id}/log-dump/obs

表 5-56 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： 获取帐号ID 、 项目ID 、 日志组ID 、 日志流ID 。 缺省值：None 最小长度：32 最大长度：32

请求参数

表 5-57 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	从IAM获取的用户Token。 缺省值：None 最小长度：1000 最大长度：2000
Content-Type	是	String	该字段填为：application/json;charset=utf8。 缺省值：None 最小长度：29 最大长度：29

表 5-58 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。 最小长度：36 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 枚举值: <ul style="list-style-type: none">7bb6b1e7-xxxx-4255-87f9-b3dc7fb2xxxx
obs_bucket_name	是	String	obs桶名称。 最小长度：3 最大长度：63
type	是	String	周期性转储, 必须填cycle。 最小长度：5 最大长度：5
storage_format	是	String	转储格式RAW/JSON, 默认为RAW。 最小长度：3 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。 最小长度：0 最大长度：64
dir_prefix_name	否	String	自定义文件夹路径。 最小长度：0 最大长度：64
period	是	Integer	转储周期的长度 枚举值: 1,2,3,5,6,12,30
period_unit	是	String	转储周期的单位: 枚举值: "min","hour" 最小长度：3 最大长度：4 说明 period与period_unit拼接后必须在该列表中 ["2min","5min","30min","1hour","3hour","6hour","12hour"]

响应参数

状态码为 201 时:

表 5-59 响应 Body 参数

参数	参数类型	描述
log_dump_objects_id	String	转储ID。 缺省值: None 最小长度: 36 最大长度: 36

状态码为 400 时:

表 5-60 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.0007
error_msg	String	调用失败响应信息描述。 枚举值: <ul style="list-style-type: none">The request body format must be json

状态码为 403 时:

表 5-61 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

状态码为 500 时:

表 5-62 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

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

状态码为 200 时:

- 获取日志组。

```
{
  "error_code": "LTS.0201",
  "error_msg": "The log group does not existed"
}
```
- 获取日志流。

```
{
  "error_code": "LTS.0208",
  "error_msg": "Log stream id does not exist: 632b9bdc-5afd-4666-a5de-2579f8b80314-"
}
```
- 设置转储OBS桶。

```
{
  "error_code": "LTS.0416",
  "error_msg": "obs bucket does not exist: zhuanchu"
}
```
- 日志流ID被关联。

```
{
  "error_code": "LTS.0207",
  "error_msg": "Log stream id is associated by transfer: 632b9bdc-5afd-4666-a5de-2579f8b80314"
}
```
- 设置转储类型。

```
{
  "error_code": "LTS.1901",
```

- ```
"error_msg": "type is not in the list [cycle]"
}
```
- 设置转储格式。

```
{
 "error_code": "LTS.1901",
 "error_msg": "storage_format is not in the list [RAW, JSON]"
}
```
  - 设置转储周期。

```
{
 "error_code": "LTS.1901",
 "error_msg": "period+period_unit is not in the list [2min, 5min, 30min, 1hour, 3hour, 6hour, 12hour]"
}
```
  - 设置转储单位。

```
{
 "error_code": "LTS.1901",
 "error_msg": "period_unit is not in the list [min, hour]"
}
```
  - 设置转储日志文件前缀。

```
{
 "error_code": "LTS.1902",
 "error_msg": "prefix_name is invalid, please verify if it's provided as required"
}
```
  - 设置自定义转储路径。

```
{
 "error_code": "LTS.1902",
 "error_msg": "dir_prefix_name is invalid, please verify if it's provided as required"
}
```

#### 状态码为 201 时:

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

#### 状态码为 400 时:

- 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"
}
```
- 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"
}
```

#### 状态码为 403 时:

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"
}
```

#### 状态码为 500 时:

- 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"
}
```

- **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。创建OBS转储成功。                                                                             |
| 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 附录

## 6.1 状态码

状态码如表1所示

表 6-1 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。    |

## 6.2 错误码

### 功能说明

所有云日志服务的接口，有自定义错误信息返回，本章节介绍云日志服务的错误码含义。

### 返回体格式

```
{
 "error_msg": "Current user is not authenticated correctly, check your token.",
 "error_code": "LTS.0002"
}
```

### 错误码说明

表 6-2 错误码

| 状态码 | 错误码      | 错误信息                                                                   | 描述                  | 处理措施                      |
|-----|----------|------------------------------------------------------------------------|---------------------|---------------------------|
| 400 | LTS.0007 | The request body format must be json                                   | 请求体不是JSON格式         | 请修改请求体为JSON格式后进行重试。       |
| 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.0101 | Failed to create log group, the group name has been existed            | 创建日志组失败，日志组名称已存在    | 核对日志组名称，此日志组名称已存在。        |
| 400 | LTS.0104 | Failed to create log group, the number of log groups exceeds the quota | 创建日志组失败，日志组数目超出限额   | 查看日志组数量是否已经达到配额（默认个数100）。 |
| 400 | LTS.0105 | Log group is associated by transfer                                    | 删除日志组失败，日志组存在关联转储配置 | 确认此组相关联的转储配置。             |
| 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。            |

| 状态码 | 错误码      | 错误信息                                                                                        | 描述                                         | 处理措施                                                     |
|-----|----------|---------------------------------------------------------------------------------------------|--------------------------------------------|----------------------------------------------------------|
| 400 | LTS.0205 | The log stream name has been existed                                                        | 创建日志流失败，日志流名称已存在                           | 确认创建日志流的名称是否存在。                                          |
| 400 | LTS.0206 | Failed to create log stream, the number of Log streams exceeds the quota                    | 创建日志流失败，单个日志组的日志流数目超出限额                    | 查看单个日志组的日志流数量是否已经达到配额（默认个数100）。                          |
| 400 | LTS.0207 | Log stream is associated by transfer                                                        | 删除日志流失败，日志流存在关联转储配置                        | 确认此日志流是否配置转储，若已配置则日志流无法删除。                               |
| 400 | LTS.0208 | The log stream does not existed                                                             | 日志流不存在                                     | 请确认要删除的日志流是否存在                                           |
| 400 | LTS.0416 | obs bucket does not exist                                                                   | 该obs桶不存在                                   | 请先创建obs桶。                                                |
| 400 | LTS.0701 | end_time must be superior to start_time, and line_num must be between them if it's provided | 时间的参数必须满足：<br>start_time<line_num<end_time | 请修改参数以满足：<br>时间的参数必须满足：<br>start_time<line_num<end_time。 |
| 400 | LTS.1901 | field is not in the list                                                                    | 某个参数不在指定的list中                             | 请根据error_msg修改对应的参数。                                     |
| 400 | LTS.1902 | name is invalid                                                                             | 某个参数不符合要求                                  | 请根据error_msg修改对应的参数。                                     |
| 400 | LTS.1903 | limit must be between lower boundary and upper boundary                                     | limit的大小不在要求范围内                            | 请将limit参数设置在1-5000之间。                                    |
| 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。                                     |
| 401 | LTS.0001 | Invalid projectId                                                                           | 无效的 projectId                              | 请确认你的uri中的projectId与token中的projectId一致。                  |
| 401 | LTS.0023 | Current user is suspended or restricted or unverified.                                      | 当前用户未进行实名认证，已被冻结或受限                        | 请确认已进行实名认证，该账户未欠费也未被冻结。                                  |

| 状态码 | 错误码      | 错误信息                                                            | 描述               | 处理措施                                     |
|-----|----------|-----------------------------------------------------------------|------------------|------------------------------------------|
| 401 | LTS.0003 | Incorrect IAM authentication information: decrypt token fail    | 无效的token         | 请重新获取token。                              |
| 403 | LTS.0001 | Invalid projectId                                               | 无效的ID            | 请重新获取ID。                                 |
| 403 | LTS.0018 | Current user does not have the permission to operate obs        | 当前用户没有操作该obs桶的权限 | 请获取操作该obs桶的权限后进行重试。                      |
| 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.0010 | Internal Server Error                                           | 内部服务器错误          | LTS服务内部异常，此时请联系技术支持工程师处理。                |
| 500 | LTS.0102 | Failed to create log group                                      | 创建日志组失败          | 核对projectID是否正确，日志组名称是否满足要求。             |
| 500 | LTS.0103 | Failed to delete log group                                      | 删除日志组失败          | 查看数据库服务是否正常或网络连接是否正常。                    |
| 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                        |
| 500 | LTS.0202 | Failed to create log stream                                     | 创建日志流失败          | 核对projectID是否正确，groupID是否正确，日志流名称是否满足要求。 |
| 500 | LTS.0203 | Failed to delete log stream                                     | 删除日志流失败          | 查看数据库服务是否正常或网络连接是否正常。                    |

## 6.3 获取帐号 ID、项目 ID、日志组 ID、日志流 ID

### 获取帐号 ID 和项目 ID

在调用接口的时候，部分URL中需要填入帐号ID（domain-id）和项目ID，获取步骤如下：

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

- 控制台获取
  - a. 登录LTS管理控制台。
  - b. 在“日志管理”页签，查看日志组ID。
  - c. 单击日志组名称，进入日志流管理页面，查看日志流ID。
- API获取
  - [查询帐号下所有日志组](#)
  - [查询指定日志组下的所有日志流](#)