

Log Tank Service

API Reference

Issue 01
Date 2024-03-05



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Before You Start	1
2 API Overview	2
3 Calling APIs	3
3.1 Obtaining Request Authentication Information	3
4 API Calling Examples	5
5 APIs	6
5.1 Log Group Management	6
5.1.1 Creating a Log Group	6
5.1.2 Querying All Log Groups of an Account	11
5.1.3 Deleting a Log Group	14
5.2 Log Stream Management	18
5.2.1 Creating a Log Stream	18
5.2.2 Querying All Log Streams in a Specified Log Group	23
5.2.3 Deleting a Log Stream	27
5.3 Log Transfer	31
5.3.1 Creating a Log Transfer Task	31
6 Appendix	38
6.1 Status Codes	38
6.2 Error Codes	39
6.3 Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID	43
A Change History	44

1 Before You Start

This document describes the functions, syntax, parameters, and examples of the Log Tank Service (LTS) APIs. [Table 1](#) lists the overall structure of this document.

Table 1-1 Document structure

Section	Description
API Overview	List of LTS APIs
Obtaining Request Authentication Information	Preparations before using APIs
API Calling Examples	Examples of the Representational State Transfer (REST) APIs to show you the API message bodies and calling methods
Log Group Management	APIs for managing LTS log groups
Log Stream Management	APIs for managing LTS log streams
Log Transfer	API for log transfer
Status Codes and Error Codes	Status codes and error codes of LTS APIs

2 API Overview

LTS extension APIs allow you to use the basic LTS functions. For example, you can query API versions, create, query, and delete log groups or log streams.

Table 1 describes the LTS APIs.

Table 2-1 API description

Category	Description
Log group management	APIs for creating, querying, and deleting log groups
Log stream management	APIs for creating, querying, and deleting log streams
Log transfer	APIs for transferring logs of one or more specified log streams to other services

3 Calling APIs

3.1 Obtaining Request Authentication Information

You can use either of the following authentication methods when calling APIs:

- Token authentication: Requests are authenticated using tokens.
- AK/SK authentication: Requests are authenticated by encrypting the request body using an Access Key ID/Secret Access Key (AK/SK) pair. AK/SK authentication is recommended because it is more secure than token authentication.

Token Authentication

For token authentication, you must obtain a token and add the **X-Auth-Token** to the request header when calling APIs.

- Step 1** Send **POST** https://IAM_endpoint/v3/auth/tokens. Obtain the Identity and Access Management (IAM) endpoint and region name in the message body.

 **NOTE**

Content in *italics* in the following example must be replaced with actual content.

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

```
}  
}  
}
```

Step 2 Obtain the token. After the request is processed, the value of **X-Subject-Token** in the message header is the token value.

Step 3 To call a service API, add **X-Auth-Token** to the request header. The value of **X-Auth-Token** is that of the token obtained in [Step 2](#).

----End

AK/SK Authentication

NOTE

AK/SK authentication supports API requests with a body no larger than 12 MB. For API requests with a larger body, use token authentication.

In AK/SK authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- **AK:** access key ID. It is a unique identifier associated with a secret access key and is used in conjunction with a secret access key to sign requests cryptographically.
- **SK:** secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or use the signing SDK to sign requests.

NOTICE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

4 API Calling Examples

This section describes how to create a log group by calling APIs.

NOTE

The token obtained from Identity and Access Management (IAM) is valid for only 24 hours. If you want to use the same token for authentication, cache it to avoid frequent calling of the IAM API.

Involved APIs

If you use a token for authentication, you must obtain the token and add **X-Auth-Token** to the request header when making an API call.

- IAM API used to obtain the token
- LTS API used to create a log group

Procedure

1. Obtain the token by referring to [Obtaining Request Authentication Information](#).
2. Send **POST /v2/{project_id}/groups**.
3. Add **Content-Type** and **X-Auth-Token** to the request header.
4. Specify the following parameters in the request body:

```
POST /v2/{project_id}/groups
{
  "log_group_name":"test001", //Log group name (The parameter is mandatory and its value is a string.)
  "ttl_in_days":"7", //Log expiration time (The value is an integer. Retain the default value.)
}
```

If the request is successful, information about the created log group is returned.

```
{
  "log_group_id":"2a0089e4-3001-11e9-9e9d-286ed48xxx", //Log group ID (The value is a string.)
}
```

If the request fails, an error code and error description are returned. For details, see [Error Codes](#).

5 APIs

All API URIs described in this chapter are case-sensitive.

5.1 Log Group Management

5.1.1 Creating a Log Group

Function

This API is used to create a log group for log retention and query. You can create a maximum of 100 log groups.

URI

POST /v2/{project_id}/groups

Table 5-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters

Request Parameters

Table 5-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Length: 30 characters

Table 5-3 Request body parameters

Parameter	Mandatory	Type	Description
log_group_name	Yes	String	Name of the log group to be created. The configuration rules are as follows: <ul style="list-style-type: none"> Must be a string of 1 to 64 characters. Only allows uppercase and lowercase letters, digits, underscores (_), hyphens (-), and periods (.). The name cannot start with a period or underscore, or end with a period. Minimum length: 1 character Maximum length: 64 characters Enumerated value: <ul style="list-style-type: none"> lts-group-01nh
ttl_in_days	Yes	Integer	Log retention duration, in days.

Response Parameters

Status code: 201

Table 5-4 Response body parameter

Parameter	Type	Description
log_group_id	String	ID of the created log group. Value length: 36 characters

Status code: 400

Table 5-5 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 401

Table 5-6 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 403

Table 5-7 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Status code: 500

Table 5-8 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Status code: 503

Table 5-9 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Example Request

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

Example Response

Status code: 201

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

Status code: 400

The request is invalid. Modify the request based on the description in **error_msg** before a retry.

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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

The server has received the request but encountered an internal error.

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

Status Codes

Status Code	Description
201	The request has succeeded and the log group has been created.
400	The request is invalid. Modify the request based on the description in error_msg before a retry.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.1.2 Querying All Log Groups of an Account

Function

This API is used to query all log groups of an account.

URI

GET /v2/{project_id}/groups

Table 5-10 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters

Request Parameters

Table 5-11 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Minimum length: 30 characters Maximum length: 30 characters

Response Parameters

Status code: 200

Table 5-12 Response body parameter

Parameter	Type	Description
log_groups	Array of LogGroup objects	Information of log groups.

Table 5-13 LogGroup

Parameter	Type	Description
creation_time	long	Time when a log group was created.
log_group_name	String	Log group name. Minimum length: 1 character Maximum length: 64 characters
log_group_id	String	Log group ID. Value length: 36 characters
ttl_in_days	Integer	Log retention duration, in days.
tag	Map<String,String>	Log group tag.

Status code: 401

Table 5-14 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Status code: 403

Table 5-15 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 500

Table 5-16 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Example Request

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

Example Response

Status code: 200

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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.


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

Status code: 500

The server has received the request but encountered an internal error.

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

Status Codes

Status Code	Description
200	The request is successful.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.1.3 Deleting a Log Group

Function

This API is used to delete a specified log group. If the log streams in a log group have been associated with log transfer tasks, you need to delete the tasks first.

URI

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

Table 5-17 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters
log_group_id	Yes	String	Log group ID. For details about how to obtain a log group ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 36 characters

Request Parameters

Table 5-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Minimum length: 30 characters Maximum length: 30 characters

Response Parameters

Status code: 400

Table 5-19 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 401

Table 5-20 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 403

Table 5-21 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 500

Table 5-22 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none">• LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none">• Invalid projectId

Example Request

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

Example Response

Status code: 400

The request is invalid. Modify the request based on the description in **error_msg** before a retry.

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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

The server has received the request but encountered an internal error.

```
{
  "error_code" : "LTS.0103",
  "error_msg" : "Failed to delete log group"
}
```

Status Codes

Status Code	Description
204	The request has succeeded and the log group has been deleted.
400	The request is invalid. Modify the request based on the description in error_msg before a retry.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.2 Log Stream Management

5.2.1 Creating a Log Stream

Function

This API is used to create a log stream in a specified log group.

URI

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

Table 5-23 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters

Parameter	Man dator y	Type	Description
log_group_id	Yes	String	ID of the log group to which the log stream to be created will belong. Default value: None Value length: 36 characters

Request Parameters

Table 5-24 Request header parameters

Parameter	Man dator y	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Minimum length: 30 characters Maximum length: 30 characters

Table 5-25 Request body parameter

Parameter	Man dator y	Type	Description
log_stream_name	Yes	String	Name of the log stream to be created. Minimum length: 1 character Maximum length: 64 characters Enumerated value: <ul style="list-style-type: none"> • lts-stream-13ci

Response Parameters

Status code: 201

Table 5-26 Response body parameter

Parameter	Type	Description
log_stream_id	String	ID of the created log stream. Value length: 36 characters

Status code: 400

Table 5-27 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 401

Table 5-28 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 403

Table 5-29 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Status code: 500

Table 5-30 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> Invalid projectId

Status code: 503

Table 5-31 Response body parameter

Parameter	Type	Description
-	String	The requested service is unavailable.

Example Request

```
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"
}
```

Example Response

Status code: 201

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

Status code: 400

The request is invalid. Modify the request based on the description in **error_msg** before a retry.


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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

The server has received the request but encountered an internal error.

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

Status Codes

Status Code	Description
201	The request has succeeded and the log stream has been created.
400	The request is invalid. Modify the request based on the description in error_msg before a retry.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.2.2 Querying All Log Streams in a Specified Log Group

Function

This API is used to query information about all log streams in a specified log group.

URI

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

Table 5-32 URI parameters

Parameter	Man dator y	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters
log_group_id	Yes	String	ID of the log group whose log streams will be queried. Default value: None Value length: 36 characters

Request Parameters

Table 5-33 Request header parameters

Parameter	Man dator y	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Minimum length: 30 characters Maximum length: 30 characters

Response Parameters

Status code: 200

Table 5-34 Response body parameter

Parameter	Type	Description
log_streams	Array of LogStream objects	List of log streams.

Table 5-35 LogStream

Parameter	Type	Description
creation_time	long	Creation time. Minimum value: 1577808000000 Maximum value: 4102416000000
log_stream_name	String	Log stream name. Value length: 36 characters
log_stream_id	String	Log stream ID. Value length: 36 characters
filter_count	Integer	Number of filters. Minimum value: 0 Maximum value: 5
tag	Map<String,String>	Log stream tag.
is_favorite	boolean	Indicates whether to add a log stream to favorites.

Status code: 401

Table 5-36 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> LTS.0403

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 403

Table 5-37 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 500

Table 5-38 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 503

Table 5-39 Response body parameter

Parameter	Type	Description
-	String	The requested service is unavailable.

Example Request

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

Example Response

Status code: 200

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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

The server has received the request but encountered an internal error.

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

Status Codes

Status Code	Description
200	The request is successful.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.2.3 Deleting a Log Stream

Function

This API is used to delete a specified log stream from a specified log group. If a log transfer task has been associated with the log stream, delete the task first.

URI

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

Table 5-40 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters
log_group_id	Yes	String	ID of the log group whose log streams will be deleted. Default value: None Value length: 36 characters
log_stream_id	Yes	String	ID of the log stream to be deleted. For details about how to obtain the log stream ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 36 characters

Request Parameters

Table 5-41 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Value length: 30 characters

Response Parameters

Status code: 400

Table 5-42 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 401

Table 5-43 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 403

Table 5-44 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 500

Table 5-45 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: <ul style="list-style-type: none"> • LTS.0403
error_msg	String	Error message. Enumerated value: <ul style="list-style-type: none"> • Invalid projectId

Status code: 503

Table 5-46 Response body parameter

Parameter	Type	Description
-	String	The requested service is unavailable.

Example Request

```
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}
```

Example Response

Status code: 400

The request is invalid. Modify the request based on the description in **error_msg** before a retry.

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

Status code: 401

Authentication failed. Check the token and try again.

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

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

The server has received the request but encountered an internal error.

```
{
  "error_code" : "LTS.0203",
  "error_msg" : "Failed to delete Log stream"
}
```

Status Codes

Status Code	Description
204	The request has succeeded and the log stream has been deleted.
400	The request is invalid. Modify the request based on the description in error_msg before a retry.
401	Authentication failed. Check the token and try again.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.

Status Code	Description
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

5.3 Log Transfer

5.3.1 Creating a Log Transfer Task

Function

This API is used to transfer logs of one or more specified log streams to Object Storage Service (OBS).

URI

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

Table 5-47 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the ID, see Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID . Default value: None Value length: 32 characters

Request Parameters

Table 5-48 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Default value: None Minimum length: 1000 characters Maximum length: 2000 characters
Content-Type	Yes	String	Set this parameter to application/json;charset=UTF-8 . Default value: None Minimum length: 30 characters Maximum length: 30 characters

Table 5-49 Request body parameters

Parameter	Mandatory	Type	Description
log_group_id	Yes	String	Log group ID. Value length: 36 characters
log_stream_ids	Yes	Array of strings	Indicates IDs of log streams whose logs are to be periodically transferred to OBS. You can specify one or more log streams. Example value: <ul style="list-style-type: none"> 7bb6b1e7-xxxx-4255-87f9-b3dc7fb2xxxx
obs_bucket_name	Yes	String	Indicates the name of an OBS bucket. Minimum length: 3 characters Maximum length: 63 characters
type	Yes	String	Set this parameter to cycle , which indicates that the log transfer is periodic. Value length: 5 characters

Parameter	Mandatory	Type	Description
storage_format		String	Indicates whether the logs are stored in raw or JSON format. The default value is RAW . Minimum length: 3 characters Maximum length: 4 characters
switch_on	No	Boolean	Indicates whether the log transfer is enabled. The value is true (default) or false .
prefix_name	No	String	Indicates the file name prefix of the log files transferred to an OBS bucket. Minimum length: 0 characters Maximum length: 64 characters
dir_prefix_name	No	String	Indicates a custom path to store the log files. Minimum length: 0 characters Maximum length: 64 characters
period	Yes	Integer	Indicates the length of the log transfer interval. Example values: 1, 2, 3, 5, 6, 12, and 30
period_unit	Yes	String	Indicates the unit of the log transfer interval. Example values: min and hour Minimum length: 3 characters Maximum length: 4 characters NOTE The log transfer interval is specified by the combination of the values of period and period_unit , and must be set to one of the following: 2 min, 5 min, 30 min, 1 hour, 3 hours, 6 hours, and 12 hours.

Response Parameters

Status code: 201

Table 5-50 Response body parameter

Parameter	Type	Description
log_dump_objects_id	String	Transfer task ID. Default value: None Value length: 36 characters

Status code: 400

Table 5-51 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value: <ul style="list-style-type: none"> LTS.0007
error_msg	String	Error message. Example value: <ul style="list-style-type: none"> The request body format must be json

Status code: 403

Table 5-52 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value:
error_msg	String	Error message. Example value: <ul style="list-style-type: none"> Invalid projectId

Status code: 500

Table 5-53 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value:

Parameter	Type	Description
error_msg	String	Error message. Example value:

Example Request

```
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"
}
```

Example Response

Status code: 200

- The log group does not exist.


```
{
  "error_code": "LTS.0201",
  "error_msg": "The log group does not existed"
}
```
- The log stream does not exist.


```
{
  "error_code": "LTS.0208",
  "error_msg": "Log stream id does not exist: 632b9bdc-5afd-4666-a5de-2579f8b80314-"
}
```
- The OBS bucket does not exist.


```
{
  "error_code": "LTS.0416",
  "error_msg": "obs bucket does not exist: zhuanchu"
}
```
- The log stream ID has been associated with a transfer task.


```
{
  "error_code": "LTS.0207",
  "error_msg": "Log stream id is associated by transfer: 632b9bdc-5afd-4666-a5de-2579f8b80314"
}
```
- Invalid transfer type.


```
{
  "error_code": "LTS.1901",
  "error_msg": "type is not in the list [cycle]"
}
```
- Invalid storage format.


```
{
  "error_code": "LTS.1901",
  "error_msg": "storage_format is not in the list [RAW, JSON]"
}
```
- Invalid log transfer interval.


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

```
"error_msg": "period+period_unit is not in the list [2min, 5min, 30min, 1hour, 3hour, 6hour, 12hour]"
}
```

- Invalid unit of the log transfer interval.

```
{
  "error_code": "LTS.1901",
  "error_msg": "period_unit is not in the list [min, hour]"
}
```

- Invalid file name prefix.

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

- Invalid custom path to store log files.

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

Status code: 201

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

Status code: 400

- The request is invalid. Modify the request based on the description in **error_msg** before a retry.

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

- The request is invalid. Modify the request based on the description in **error_msg** before a retry.

```
{
  "error_code": "LTS.0007",
  "error_msg": "The request body format must be json"
}
```

Status code: 403

The server understood the request but refused to authorize it. The client should not repeat the request without modifications.

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

Status code: 500

- The server has received the request but encountered an internal error.

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

- The server has received the request but encountered an internal error.

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

Status Codes

Status Code	Description
200	The request has succeeded.
201	The request has succeeded. The transfer task has been created.
400	The request is invalid. Modify the request based on the description in error_msg before a retry.
403	The server understood the request but refused to authorize it. The client should not repeat the request without modifications.
500	The server has received the request but encountered an internal error.
503	The requested service is unavailable.

Error Codes

For details, see [Error Codes](#).

6 Appendix

6.1 Status Codes

[Table 1](#) lists the status codes.

Table 6-1 Status codes

Status Code	Returned Value	Description
200	OK	The normal response for the GET or PUT operation is returned.
201	OK	The POST request is successful and the query result is returned.
204	No Content	The normal response for the DELETE operation is returned.
400	Bad Request	Request error.
401	Unauthorized	The authentication information is not provided or is incorrect.
403	Forbidden	You are forbidden to access the page requested.
404	Not Found	The server failed to find the requested resource.
408	Request Timeout	The request timed out.
429	Too Many Requests	The number of requests exceeded the upper limit.
500	Internal Server Error	The server encountered an unexpected condition which prevented it from fulfilling the request.
503	Service Unavailable	The service is currently unavailable.

6.2 Error Codes

Introduction

This section explains the meanings of error codes returned by LTS APIs.

Response Format

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

Error Code Description

Table 6-2 Error codes

Stat us Code	Error Code	Error Message	Description	Solution
400	LTS.0007	The request body format must be json	The request body is not in JSON format.	Modify the request body to the JSON format and try again.
400	LTS.0009	Failed to validate the request body	Parameter verification failed.	Modify request parameters based on the returned error information and try again.
400	LTS.0010	The system encountered an internal error	An internal error occurred.	An internal error occurred in LTS. Contact technical support.
400	LTS.0101	Failed to create log group, the group name has been existed	Failed to create the log group because a log group with the same name already exists.	Check the log group name.

Stat us Code	Error Code	Error Message	Description	Solution
400	LTS.0104	Failed to create log group, the number of log groups exceeds the quota	Failed to create the log group because the maximum number of log groups allowed has been reached.	Check whether the number of log groups has reached the quota (100 by default).
400	LTS.0105	Log group is associated by transfer	Failed to delete the log group because the log group has been associated with log transfer tasks.	Delete the associated log transfer tasks first.
400	LTS.0106	The log group TTL day must be 7	The log retention duration is seven days by default.	Ensure that the log retention duration is seven days.
400	LTS.0201	The log group is not existed	Failed to create the log stream because the associated log group does not exist.	Check whether the ID of the log group is correct.
400	LTS.0205	The log stream name has been existed	Failed to create the log stream because a log stream with the same name already exists.	Check the log stream name.
400	LTS.0206	Failed to create log stream, the number of Log streams exceeds the quota	Failed to create the log stream because the maximum number of log streams allowed has been reached.	Check whether the number of log streams in the log group has reached the quota (100 by default).
400	LTS.0207	Log stream is associated by transfer	Failed to delete the log stream because the log stream has been associated with log transfer tasks.	Check whether the log stream is associated with a log transfer task. A log stream associated with a log transfer task cannot be deleted.

Stat us Code	Error Code	Error Message	Description	Solution
400	LTS.0208	The log stream does not exist	The log stream does not exist.	Check whether the log stream to be deleted exists.
400	LTS.0416	obs bucket does not exist	The OBS bucket does not exist.	Create an OBS bucket first.
400	LTS.0701	end_time must be superior to start_time, and line_num must be between them if it's provided	The values of time parameters must meet the following requirements: start_time < line_num < end_time	Modify the parameter values to meet the requirements.
400	LTS.1901	field is not in the list	A parameter is not in the specified list.	Modify the parameter based on the description in error_msg .
400	LTS.1902	name is invalid	A parameter is invalid.	Modify the parameter based on the description in error_msg .
400	LTS.1903	limit must be between lower boundary and upper boundary	The value of limit is not in the range allowed.	Set limit to a value ranging from 1 to 5000.
400	LTS.0301	'*' and '?' not allowed as first character	Asterisks (*) and question marks (?) are placed in the middle or at the end of a keyword.	Check the keywords field based on the error information.
401	LTS.0001	Invalid projectId	Invalid project ID.	Ensure that the project ID in the URI is the same as that in the token.
401	LTS.0023	Current user is suspended or restricted or unverified.	Your account has not completed the real-name authentication, or has been frozen or restricted.	Ensure that your account has completed the real-name authentication, and is not in arrears or frozen.

Stat us Code	Error Code	Error Message	Description	Solution
401	LTS.0003	Incorrect IAM authentication information: decrypt token fail	Invalid token.	Obtain a token again.
403	LTS.0001	Invalid projectId	Invalid ID.	Obtain the ID again.
403	LTS.0018	Current user does not have the permission to operate obs	You do not have the permissions required to perform operations on the OBS bucket.	Obtain the permissions required and try again.
403	LTS.0011	Invalid projectId	Invalid project ID.	Ensure that the project ID in the URL is the same as that in the token.
500	LTS.0010	Internal Server Error	Internal server error.	An internal error occurred in LTS. Contact technical support.
500	LTS.0102	Failed to create log group	Failed to create the log group.	Check whether the project ID is correct and whether the log group name meets the requirements.
500	LTS.0103	Failed to delete log group	Failed to delete the log group.	Check whether the database or network connection is normal.
500	LTS.0107	Current user does not have the permission to operate this group	Failed to update the log group.	Abnormal connection to the database. Check the database instance status.
500	LTS.0202	Failed to create log stream	Failed to create the log stream.	Check whether the project ID and group ID are correct and whether the log stream name meets the requirements.

Status Code	Error Code	Error Message	Description	Solution
500	LTS.0203	Failed to delete log stream	Failed to delete the log stream.	Check whether the database or network connection is normal.

6.3 Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID

Obtaining Account and Project IDs

Account and project IDs are required in URLs when you call some APIs.

1. Log in to the management console.
2. Hover the mouse pointer over the username and choose **My Credentials** from the drop-down list.
3. On the **My Credentials** page, obtain the account ID and project ID.

Obtaining Log Group and Log Stream IDs

- Obtain the IDs from the console.
 - a. Log in to the LTS console.
 - b. On the **Log Management** page, obtain the log group ID.
 - c. Click the name of a log group and obtain the log stream ID on the page displayed.
- Obtain the IDs by calling APIs.
 - [Querying All Log Groups of an Account](#)
 - [Querying All Log Streams in a Specified Log Group](#)

A Change History

Released On	Description
2021-03-30	Added the following API: Creating a Log Transfer Task
2019-03-30	This issue is the first official release.