Log Tank Service

API Reference

Issue 01

Date 2023-11-29

Contents

1 Before You Start

1.1 Overview

Log Tank Service (LTS) collects log data from hosts and cloud services. By processing massive amounts of log data efficiently, securely, and in real time, LTS provides useful insights for you to optimize the availability and performance of cloud services and applications. Using LTS, you can efficiently perform real-time decision-making, device O&M management, and service trend analysis.

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

1.2 API Calling

LTS supports Representational State Transfer (REST) APIs that can be called over HTTPS. For details about API calling, see section *Calling APIs*.

1.3 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoints of all services, see **Regions and Endpoints**.

1.4 Notes and Constraints

- The amount of LTS resources that you can create is determined by your quota.
- For details about the constraints, see API descriptions.

1.5 Basic Concepts

- User

An IAM user is created by an account in IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys).

The account name, username, and password will be required for API authentication.

- Region

  A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

- AZ

  An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

  Projects group and isolate resources (including compute, storage, and network resources) across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolating model
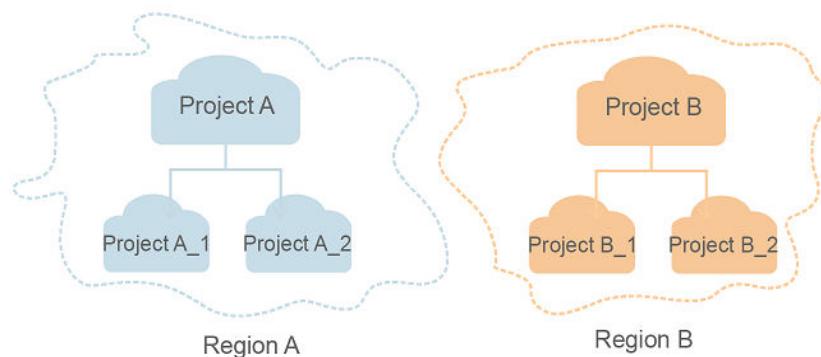
2 API Overview

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 management	APIs for querying logs
Log transfer	APIs for transferring logs of one or more specified log streams to other services

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for **obtaining a user token** as an example to demonstrate how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

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

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

- **URI-scheme**:

  Protocol used to transmit requests. All APIs use **HTTPS**.

- **Endpoint**:

  Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from **Regions and Endpoints**.

  For example, the endpoint of IAM in the AP-Kuala Lumpur region is **iam.my-kualalumpur-1.myhuaweicloud.com**.

- **resource-path**:

  Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.

- **query-string**:

  Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of "*Parameter name=Parameter value*". For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

📖 **NOTE**

> To simplify the URI display, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server:

- **GET**: requests a server to return specified resources.
- **PUT**: requests a server to update specified resources.
- **POST**: requests a server to add resources or perform special operations.
- **DELETE**: requests a server to delete specified resources, for example, objects.
- **HEAD**: same as GET except that the server must return only the response header.
- **PATCH**: requests a server to update a part of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to obtain a user token, the request method is **POST**. The request is as follows:

POST https://iam.my-kualalumpur-1.cloudalphai4.tmone.com.my/v3/auth/tokens

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Table 3-1 describes common request headers.

Table 3-1 Common request headers

Parameter	Description	Mandatory	Example Value
Host	Server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of *Hostname:Port number*. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Type (or format) of the message body. The default value **application/json** is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	No	3495
X-Project-Id	Project ID. Obtain the project ID by following the instructions in **Obtaining the AccountID, Project ID, Log Group ID, and Log Stream ID**.	No	e9993fc787d94b6c886cb aa340f9c0f4

Parameter	Description	Mandatory	Example Value
X-Auth-Token	User token. It is a response to the API for **obtaining a user token**. This API is the only one that does not require authentication. After the request is processed, the value of **X-Subject-Token** in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQc-Co...ggg1BBIINPXsidG9rZ

☐ **NOTE**

In addition to supporting token-based authentication, APIs also support authentication using access key ID/secret access key (AK/SK). During AK/SK-based authentication, an SDK is used to sign a request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

For details, see "AK/SK-based Authentication" in **Authentication**.

The API used to **obtain a user token** does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.my-kualalumpur-1.cloudalphai4.tmone.com.my/v3/auth/tokens
Content-Type: application/json
```

Request Body (Optional)

This part is optional. A request body is often sent in a structured format (for example, JSON or XML) as defined in the **Content-Type** header field.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to **obtain a user token**, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from **Regions and Endpoints**.

☐ **NOTE**

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. For details, see **Obtaining a User Token**.

```
POST https://iam.my-kualalumpur-1.cloudalphai4.tmone.com.my/v3/auth/tokens
Content-Type: application/json
```

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

If all data required for the API request is available, you can send the request to call an API through **curl**, **Postman**, or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

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

- Token-based authentication: Requests are authenticated using a token.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair. AK/SK-based authentication is recommended because it is more secure than token-based authentication.

Token-based Authentication

☐ NOTE

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token is used to acquire temporary permissions. During API authentication using a token, the token is added to requests to get permissions for calling the API.

You can obtain a token by calling the **Obtaining User Token** API. When you call the API, set **auth.scope** in the request body to **project**.

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

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

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
POST https://iam.my-kualalumpur-1.cloudalphai4.tmone.com.my/v3/auth/tokens
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

📖 NOTE

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

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

- AK: access key ID, which is a unique identifier 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-based 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.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For details, see **Error Codes**.

For example, if status code **201** is returned for calling the API used to obtain a user token, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-type**.

Figure 1 shows the response header fields for the API used to obtain a user token. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-1 Header fields of the response to the request for obtaining a user token

```
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
→   MIIYXQYJKoZIhvcNAQcCoIIYTjCCGEoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOnsiZXhwaXJlc19hdCI6IjIwMTktMDItMTNUMD
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkqjACgklqO1wi4JlGzrpd18LGXK5txldfq4lqHCYb8P4NaY0NYejcAgzJVeFIYtLWT1GSO0zxKZmlQHQj82HBqHdglZO9fuEbL5dMhdavj+33wEl
xHRCE9l87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jqglFkNPQuFSOU8+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection →   1; mode=block;
```

Response Body (Optional)

The body of a response is often returned in structured format as specified in the **Content-type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to obtain a user token.

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

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

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

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

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 **Making an API Request** .
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	Man datory	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	Man datory	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	Man datory	Type	Description
log_group_na me	Yes	String	Name of the log group to be created. The configuration rules are as follows: ● 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: ● **lts-group-01nh**
ttl_in_days	Yes	Integer	Log retention duration, in days. Maximum value: **7**

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: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 401

Table 5-6 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-7 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 500

Table 5-8 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 503

Table 5-9 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **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	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

Request Parameters

Table 5-11 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 5-12 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. Minimum value: **1** Maximum value: **7**
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: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-15 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: • **LTS.0403**
error_msg	String	Error message. Enumerated value: • **Invalid projectId**

Status code: 500

Table 5-16 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: • **LTS.0403**
error_msg	String	Error message. Enumerated value: • **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: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 401

Table 5-20 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-21 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 500

Table 5-22 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **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	Mandatory	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	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 5-25 Request body parameter

Parameter	Mandatory	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: • **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: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 401

Table 5-28 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-29 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: • **Invalid projectId**

Status code: 500

Table 5-30 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: • **LTS.0403**
error_msg	String	Error message. Enumerated value: • **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	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 queried. Default value: None Value length: 36 characters

Request Parameters

Table 5-33 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 5-34 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: ● **LTS.0403**

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: • **Invalid projectId**

Status code: 403

Table 5-37 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: • **LTS.0403**
error_msg	String	Error message. Enumerated value: • **Invalid projectId**

Status code: 500

Table 5-38 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: • **LTS.0403**
error_msg	String	Error message. Enumerated value: • **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	Man datory	Type	Description
project_id	Yes	String	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	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 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: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 401

Table 5-43 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**

Parameter	Type	Description
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-44 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 500

Table 5-45 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **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 Management

5.3.1 Querying Logs

Function

This API is used to query logs in a specified log stream.

URI

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

Table 5-47 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	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 Default value: None Value length: 36 characters
log_stream_id	Yes	String	Log stream ID Default value: None Value length: 36 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
start_time	Yes	String	UTC start time of the search window (in milliseconds). **NOTE** Maximum query time range: 30 days
end_time	Yes	String	UTC end time of the search window (in milliseconds). **NOTE** Maximum query time range: 30 days
labels	No	Map<String,String>	Filter criteria, which vary between log sources.
keywords	No	String	Keyword used for search. A keyword is a word between two adjacent delimiters. Enumerated value: ● **error**

Parameter	Man datory	Type	Description
line_num	No	String	Sequence number of a log event. This parameter is not required for the first query, but is required for subsequent pagination queries. The value can be obtained from the response of the last query. The value of **line_num** should be between the values of **start_time** and **end_time**. Value length: 19 characters
is_desc	No	Boolean	Whether the search order is descending or ascending. The default value is **false**, indicating that search results are displayed in ascending order. Enumerated value: ● **false**
search_type	No	String	The value is **init** (default value) for the first query, or **forwards** or **backwards** for a pagination query. This parameter is used together with **is_desc** for pagination queries. Enumerated value: ● **forwards**
limit	No	Integer	Number of logs to be queried each time. The value is **50** when this parameter is not set. You are advised to set this parameter to **100**. Minimum value: **1** Maximum value: **5000**

Response Parameters

Status code: 200

Table 5-50 Response body parameter

Parameter	Type	Description
logs	Array of **LogContents** objects	Information of logs.
count	Integer	Number of logs.

Table 5-51 LogContents

Parameter	Type	Description
content	String	Raw log data. Minimum length: 1 character Maximum length: 10,000 characters
line_num	String	Sequence number of a log event. Value length: 19 characters
labels	Map<String,String>	Labels contained in a log event. The labels vary depending on log events.

Status code: 400

Table 5-52 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 401

Table 5-53 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 403

Table 5-54 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Status code: 500

Table 5-55 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Enumerated value: ● **LTS.0403**
error_msg	String	Error message. Enumerated value: ● **Invalid projectId**

Example Request

Log details:

```
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
```

● For the first query:
```
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
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 6**, **NO 7**, and **NO 8** are the target log events):

```
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
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 8**, **NO 7**, and **NO 6** are the target log events):

```
{
  "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
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 2**, **NO 3**, and **NO 4** are the target log events):

```
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
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 4**, **NO 3**, and **NO 2** are the target log events):

```
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
        }
```

Example Response

- For the first query:

```
{
 "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"
               }
            ]
         }
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 6**, **NO 7**, and **NO 8** are the target log events):

```
         {
            "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"
    }
    ]
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 8**, **NO 7**, and **NO 6** are the target log events):

```
{
    "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"
      }
   ]
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 2**, **NO 3**, and **NO 4** are the target log events):

```
{
    "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"
        }
    ]
}
```

- For a pagination query (Assume that the search starts from the log event containing **NO 5** and the log events containing **NO 4**, **NO 3**, and **NO 2** are the target log events):

```
{
    "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"
        }
    ]
}
```

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 query lts log"
}
```

Status Codes

Status Code	Description
200	The request is successful.
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**.

5.4 Log Transfer

5.4.1 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 5-56 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 5-57 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 5-58 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: ● 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
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**

Parameter	Man datory	Type	Description
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 5-59 Response body parameter

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

Status code: 400

Table 5-60 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value: LTS.0007
error_msg	String	Error message. Example value: The request body format must be json

Status code: 403

Table 5-61 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value:
error_msg	String	Error message. Example value: ● Invalid projectId

Status code: 500

Table 5-62 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Example value:
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

Status 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.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.
400	LTS.0208	The log stream does not existed	The log stream does not exist.	Check whether the log stream to be deleted exists.

Stat us Code	Error Code	Error Message	Description	Solution
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.

If there are multiple projects in one region, expand **Region** and obtain subproject IDs from the **Project ID** column.

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**