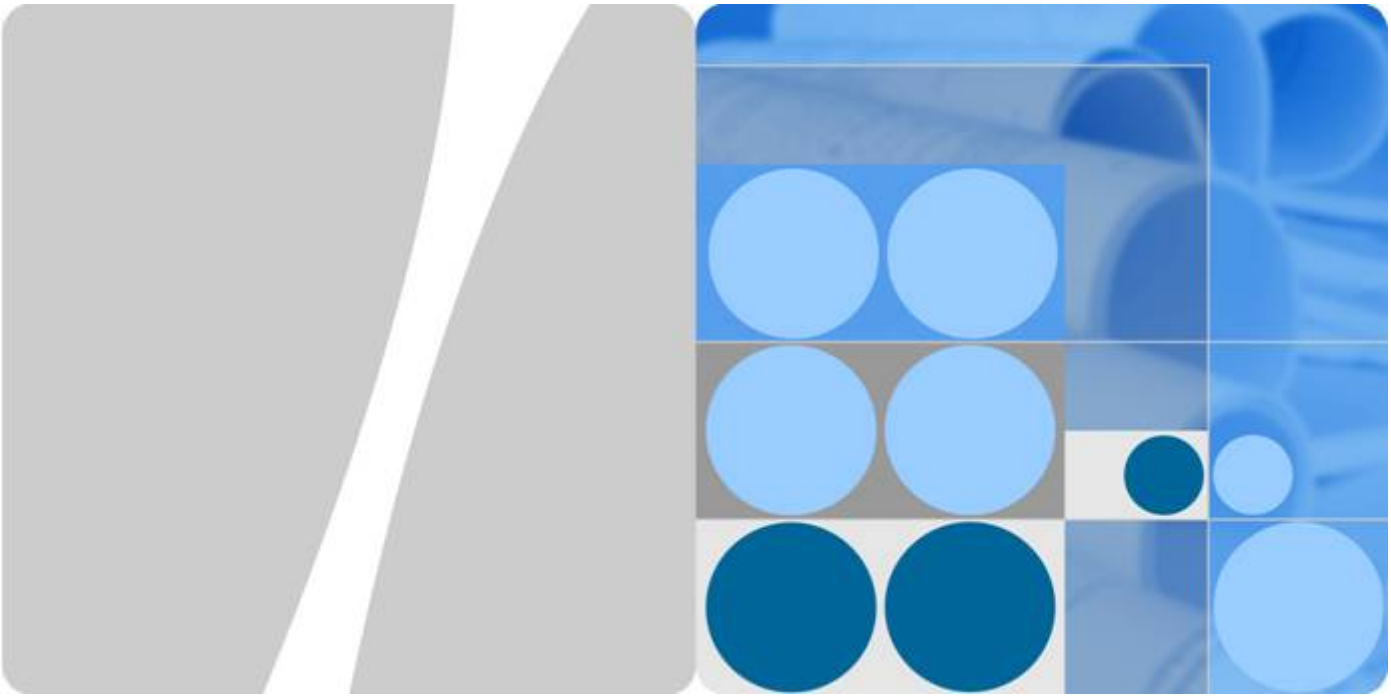


Cloud Trace Service

API Reference

Date **2019-09-28**

Contents

1 Before You Start

Overview

Cloud Trace Service (CTS) is a log audit service that is available for cloud security. It allows you to collect, store, and query resource operation records. You can use these records to perform security analysis, track resource changes, audit compliance, and locate faults.

You can use APIs provided in this document to perform operations on the CTS service, such as creating and deleting a tracker. Before calling the APIs of the CTS service, ensure that you are familiar with related concepts and functions of the CTS service.

Endpoints

An endpoint is a part of API information. Endpoints vary depending on services and regions. For the endpoints of CTS, see **Regions and Endpoints**.

2 API Overview

Cloud Trace Service (CTS) provides extension APIs. CTS APIs allow you to use all CTS functions. For example, query the API version or trace list or create a tracker.

Table 1 lists APIs provided by CTS.

Table 2-1 CTS APIs

Subtype	Description
API version	API for querying version information of all CTS APIs or a specified API
Tracker	API for creating, modifying, querying, or deleting a tracker
Trace	API for querying traces recorded in the past 7 days

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a Representational State Transfer (REST) API request, and uses the Identity and Access Management (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 endpoint. The endpoint varies depending on the service and service region.

  For example, the endpoint of IAM in the **CN North-Beijing1** region is **iam.cn-north-1.myhuaweicloud.com**.

- **resource-path**:

  Resource path, that is, API access path. 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**:

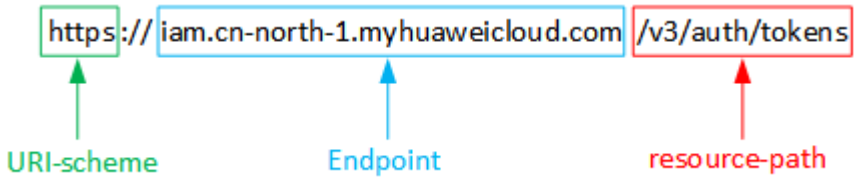
  Query parameter, which is optional. Not all APIs have a query parameter. Ensure that a **question mark (?)** is included before each query parameter that is in the format of **Parameter name=Parameter value**. For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **CN North-Beijing1** region, obtain the endpoint of IAM (**iam.cn-north-1.myhuaweicloud.com**) for this region and the

resource-path (**/v3/auth/tokens**) in the URI of the API used to obtain a user token. Then, construct the URI as follows:

```
https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, 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 the server to return specified resources.
- **PUT**: requests the server to update specified resources.
- **POST**: requests the server to add resources or perform special operations.
- **DELETE**: requests the server to delete specified resources, for example, an object.
- **HEAD**: same as GET except that the server must return only the response header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource can be created using the PATCH method.

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.cn-north-1.myhuaweicloud.com/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.

Common request header fields are as follows:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. User token is a response to the API for obtaining a user token (only this API does not require authentication).

> **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 the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

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.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body

The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header. If the request body contains Chinese characters, these characters must be coded in UTF-8.

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. Add the message body to the following request. The italic fields in bold must be set based on the site requirements. In the request, ***username*** indicates the IAM username, ***domainname*** indicates the IAM user account name, ************* indicates the login password, and ***xxxxxxxxxxxxxxxxx*** indicates the project name, for example, **cn-north-1**.

> **NOTE**
>
> The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see obtaining a user token API.

```
POST https://iam.cn-north-1.myhuaweicloud.com/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 the 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 Access Key ID/Secret Access Key (AK/SK) pair. AK/SK-based authentication is recommended because it provides higher security than token authentication.

Token Authentication

NOTE

The validity period of a token is 24 hours. If the same token is used for authentication, cache it to prevent frequent API calls.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.

The process of calling the API for obtaining a user token is described as an example. 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.cn-north-1.myhuaweicloud.com/v3/auth/projects
POST https://iam.eu-west-0.prod-cloud-ocb.orange-business.com/v3/auth/projects
POST https://iam.sa-brazil-1.telefonicaopencloud.com/v3/auth/projects
POST https://{{endpoint}}/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

NOTE

AK/SK-based authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token-based authentication is recommended.

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. 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.

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 Code**.

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 3-2 Header fields of the response to the request for obtaining a user token

Response Body

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**. The following shows part of the response body for the API to obtain a user token.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "cn-north-1",
.....
```

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 Getting Started

This section describes how to create a tracker by calling Cloud Trace Service (CTS) APIs.

NOTE

The token obtained from Identity and Access Management (IAM) is valid for only 24 hours. If you want to use one token for authentication, you can cache it to avoid frequently obtaining the token.

Involved APIs

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

- API for obtaining tokens from IAM
- API for creating a tracker

Procedure

1. Obtain the token by referring to **Authentication**.
2. Send **POST /v1.0/ {project_id} /tracker**.
3. Add **Content-Type** and **X-Auth-Token** to the request header.
4. Specify the following parameters in the request body:

```
POST /v1.0/{project_id}/tracker
{
 "bucket_name": "obs-f1da",  // Select the name of the OBS bucket to be dumped. This parameter is mandatory and its value is String.
  "is_support_trace_files_encryption": true,  // Specifies whether to enable the encryption function for trace files to be transferred. This parameter is mandatory and its value is Boolean.
 "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",  // Specifies the key ID for encrypting trace files to be transferred. When is_support_trace_files_encryption is set to true, this parameter is mandatory and its value is Boolean.
 "is_obs_created": true,  //Specifies whether to create an OBS bucket. This parameter is optional and its value is Boolean.
 "file_prefix_name": "yO8Q",   // Specifies the prefix of the log file to be stored in OBS. This parameter is optional and its value is String.
  "log_file_validate": {
 "is_support_validate": true  // Specifies whether to enable event file verification. This parameter is optional and its value is Boolean.
 }
}
```

After the request is successfully processed, information about the created tracker is returned.

```
{
 "bucket_name": "obs-f1da",
 "file_prefix_name": "yO8Q",
 "is_obs_created": true,
 "is_support_trace_files_encryption": true,
 "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",
  "log_file_validate": {
 "is_support_validate": true
 },
Specifies the tracker name,  // Identifies the tracker name.
"Status": "Enabled"  // Identifies the tracker status.
}
```

If the request fails, the error code and information are returned. For details, see **Error Code**.

5 API

5.1 API Version Management

5.1.1 Querying All API Versions

Function

This API is used to query all API versions supported by CTS.

URI

GET /

Request

None

Response

- Parameter description

Table 5-1 Parameters in the response

Parameter	Mandatory	Type	Description
version	Yes	Array	Specifies the list of all API versions.
id	Yes	String	Specifies the version ID (version number), for example, v1.
links	Yes	String	Specifies the API URL.
href	Yes	String	Specifies the reference address of the current API version.

Parameter	Mandatory	Type	Description
rel	Yes	String	Specifies the relationship between the current API version and the referenced address.
version	Yes	String	If the APIs of this version support microversions, set this parameter to the supported maximum microversion. If the microversion is not supported, leave this parameter empty.
status	Yes	String	Specifies the version status. The options are as follows: **CURRENT**: indicates that the version is the primary version. **SUPPORTED**: indicates that the version is an old version, but it is still supported. **DEPRECATED**: indicates that the version is a deprecated version, which may be deleted later.
updated	Yes	String	Specifies the version release time, which must be the UTC time. For example, the release time of v1 is 2014-06-28T12:20:21Z.
min_version	No	String	If the APIs of this version support microversions, set this parameter to the supported minimum microversion. If the microversion is not supported, leave this parameter empty.

- Example response

```
{
 "version": [
  {
    "id": "v1.0",
    "links": [
     {
       "href": "https://x.x.x.x/v1.0/",
       "rel": "self"
     }
    ],
    "min_version": "",
    "status": "CURRENT",
    "updated": "2018-09-30T00:00:00Z",
    "version": ""
  }
 {
    "id": "v2.0",
    "links": [
     {
       "href": "https://x.x.x.x/v2.0/",
```

```
        "rel": "self"
      }
    ],
    "min_version": "",
    "status": "SUPPORTED",
    "updated": "2018-09-30T00:00:00Z",
    "version": ""
  }
 ]
}
```

Returned Value

- Normal

  200
- Abnormal

Table 5-2 Return code for failed requests

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	The request was forbidden.
404 Not Found	The server failed to find the requested resource.
408 Request Timeout	The request timed out.
429 Too Many Requests	The number requests exceeded the upper limit.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

5.1.2 Querying a Specified API Version

Function

This API is used to query a specified API version of Cloud Trace Service (CTS).

URI

GET /{api_version}

For details about the parameters, see **Table 1**.

Table 5-3 Parameters in the URI

Parameter	Mandatory	Description
api_version	Yes	API version

Request

None.

Response

- Parameter description

Table 5-4 Parameters in the request

Parameter	Mandatory	Type	Description
version	Yes	Array	List of all API versions.
id	Yes	String	Version ID (version number), for example, v1.
links	Yes	String	API URL.
href	Yes	String	Reference address of the current API version.
rel	Yes	String	Relationship between the current API version and the referenced address.
version	Yes	String	If APIs of this version support microversions, set this parameter to the supported maximum microversion. If the microversion is not supported, leave this parameter empty.
status	Yes	String	Version status. The options are as follows: **CURRENT**: The version is the primary version. **SUPPORTED**: The version is an old version, but it is still supported. **DEPRECATED**: The version is a deprecated version, which may be deleted later.

Parameter	Mandatory	Type	Description
updated	Yes	String	Version release time, which must be the UTC time. For example, the release time of v1 is 2014-06-28T12:20:21Z.
min_version	No	String	If APIs of this version support microversions, set this parameter to the supported minimum microversion. If the microversion is not supported, leave this parameter empty.

- Example response

```
{
 "version":
  {
    "id": "v1.0",
    "links": [
     {
       "href": "https://x.x.x.x/v1.0/",
       "rel": "self"
     }
    ],
    "min_version": "",
    "status": "CURRENT",
    "updated": "2018-09-30T00:00:00Z",
    "version": ""
  }
}
```

Returned Value

- Normal

  200
- Abnormal

Table 5-5 Return code for failed requests

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is unavailable or incorrect.
403 Forbidden	You are forbidden to access the requested page.
404 Not Found	The server failed to find the requested resource.
408 Request Timeout	The request timed out.

Returned Value	Description
429 Too Many Requests	The number requests exceeded the upper limit.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	Failed to complete the request because the system is unavailable.

5.2 Tracker Management

5.2.1 Creating a Tracker

All API URLs described in this section are case-sensitive.

Function

This API is used to create a tracker.

A tracker will be automatically created after Cloud Trace Service (CTS) is enabled. All traces recorded by CTS are associated with the tracker. Currently, only one tracker can be created for an account in a region.

On the management console, you can query the last seven days of operation records. To obtain more operation records, you can enable Object Storage Service (OBS) and deliver operation records to OBS buckets for long-term storage in real time.

URI

POST /v1.0/{project_id}/tracker

For details about the parameters, see **Table 1**.

Table 5-6 Parameters in the URI

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request

- Parameters

Table 5-7 Parameters in the request

Parameter	Sub-Parameter	Mandatory	Type	Description
bucket_name	N/A	Yes	String	OBS bucket name. Starts with a digit or letter and contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.)
file_prefix_name	N/A	No	String	Prefix of a log that needs to be stored in an OBS bucket. The value is a string of 0 to 64 characters and can contain uppercase and lowercase letters (a to z and A to Z), digits (0 to 9), hyphens (-), underscores (_), or periods (.)
is_obs_created	N/A	Yes	Boolean	Whether a new OBS bucket can be created. • When the value is **true**, you can create an OBS bucket to store trace files. • When the value is **false**, you can select an existing OBS bucket to store trace files. The bucket name is a string of 3 to 64 characters and can only contain digits, letters, hyphens (-), and periods (.). Only one period and hyphen can be contained in the string.
is_support_trace_files_encryption	N/A	Yes	Boolean	Whether the function of encrypting stored trace files is enabled or not. This parameter must be used with **kms_id**.

Parameter	Sub-Parameter	Mandatory	Type	Description
kms_id	N/A	Yes	String	Key ID used for dumping and encrypting trace files. This key ID is obtained from KMS. This parameter is mandatory when the value of **is_support_trace_files_encryption** is **true**.
lts	is_lts_enabled	Yes	Boolean	Whether the LTS search function is enabled or not.
	log_group_name	Yes	String	Name of the log group that CTS creates in LTS.
	log_topic_name	Yes	String	Name of the log topic that CTS creates in LTS.
log_file_validate	N/A	No	json	Whether to enable the trace file verification function.
-	is_support_validate	Yes	Boolean	Whether the trace file verification function is supported or not.

- Example request

```
POST /v1.0/{project_id}/tracker
{
    "bucket_name": "obs-f1da",
    "is_support_trace_files_encryption": true,
    "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",
    "is_obs_created": true,
    "file_prefix_name": "yO8Q",
    "lts": {
        "is_lts_enabled": true,
        "log_group_name": "CTS",
        "log_topic_name": "system-trace"
    },
    "log_file_validate": {
        "is_support_validate": true
    }
}
```

Response

- Parameters

Table 5-8 Parameters in the response

Parameter	Sub-Parameter	Type	Description
bucket_name	N/A	String	OBS bucket name. Starts with a digit or letter and contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.)
file_prefix_name	N/A	String	Prefix of a log that needs to be stored in an OBS bucket.
status	N/A	String	Status of a tracker. The value is **enabled**.
tracker_name	N/A	String	Tracker name. Currently, only tracker "system" is available.
is_obs_created	N/A	Boolean	Whether a new OBS bucket can be created. • When the value is **true**, you can create an OBS bucket to store trace files. • When the value is **false**, you can select an existing OBS bucket to store trace files. The bucket name is a string of 3 to 64 characters and can only contain digits, letters, hyphens (-), and periods (.). Only one period and hyphen can be contained in the string.

Parameter	Sub-Parameter	Type	Description
is_support_trace_files_encryption	N/A	Boolean	Whether the function of encrypting stored trace files is enabled or not. This parameter must be used with **kms_id**.
kms_id	N/A	String	Key ID used for dumping and encrypting trace files. This key ID is obtained from KMS. This parameter is mandatory when the value of **is_support_trace_files_encryption** is **true**.
lts	is_lts_enabled	Boolean	Name of the log group that CTS creates in LTS.
	log_topic_name	String	Name of the log topic that CTS creates in LTS.
	log_group_id	String	ID of the log group that CTS creates in LTS.
	log_topic_id	String	ID of the log topic that CTS creates in LTS.
log_file_validate	log_file_validate	JSON object	Whether to enable the trace file verification function.
is_support_validate	is_support_validate	Boolean	Whether the trace file verification function is supported or not.

- Example response

```
{
    "bucket_name": "obs-f1da",
    "file_prefix_name": "yO8Q",
    "is_obs_created": true,
    "is_support_trace_files_encryption": true,
    "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",
        "need_notify_user_list": [
            "user1",
            "user2"
        ]
```

```
    },
    "lts": {
        "is_lts_enabled": true,
        "log_group_name": "CTS",
        "log_topic_name": "system-trace",
        "log_group_id": "b6895f40-b897-11e7-8aa6-2866d488c81c",
        "log_topic_id": "b68d5281-b897-11e7-8aa7-2866d488c81c"
    },
    "log_file_validate": {
        "is_support_validate": true
    },
    "tracker_name": "system",
    "status": "enabled"
}
```

Returned Value

- Normal

  Table 5-9 Return code for successful requests

Returned Value	Description
201	The request is successfully processed.

- Abnormal

  Table 5-10 Return code for failed requests

Returned Value	Description
400	The server failed to process the request.
403	You are forbidden to access the requested page.
500	Failed to complete the request because of an internal service error.
401	Your access request is rejected.
404	The requested OBS bucket does not exist.

5.2.2 Modifying a Tracker

Function

Configuration items of the tracker can be modified, including Trace Transfer to OBS, Key Event Notification, Encrypt Trace File, Retrieve Management Traces Using LTS, Verify Trace File, and parameters for enabling and disabling a tracker. Modifying the tracker does not affect the existing operation records. After the modification is complete, the system will immediately start recording operations based on the new rule.

URI

PUT /v1.0/{project_id}/tracker/{tracker_name}

For details about the parameters, see **Table 1**.

Table 5-11 Parameters in the URI

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
tracker_name	Yes	String	Name of a tracker. Currently, only tracker "system" is available.

Request

- Parameters

Table 5-12 Parameters in the request

Parameter	Sub-Parameter	Mandatory	Type	Description
bucket_name	N/A	Yes	String	OBS bucket name. Starts with a digit or letter and contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.)
file_prefix_name	N/A	No	String	Prefix of a log that needs to be stored in an OBS bucket. The value is a string of 0 to 64 characters and can contain uppercase and lowercase letters (a to z and A to Z), digits (0 to 9), hyphens (-), underscores (_), or periods (.)
status	N/A	No	String	Status of a tracker. The value can be **enabled** or **disabled**. If you change the value to **disabled**, the tracker stops recording traces.

Parameter	Sub-Parameter	Mandatory	Type	Description
data_bucket	data_bucket_name	Yes	Array	Bucket name of the data class tracker. • This parameter is mandatory whenever the data class tracker is enabled or disabled. • This parameter is not invalid for the management class tracker.
	data_event	Yes	Array	Type of operations tracked by the data tracker. • This parameter is mandatory whenever the data class tracker is enabled or disabled. • This parameter is not invalid for the management class tracker.
	search_enabled	Yes	Boolean	Whether to enable trace analysis for the data tracker. • This parameter is mandatory whenever the data class tracker is enabled or disabled. • This parameter is not invalid for the management class tracker.

Parameter	Sub-Parameter	Mandatory	Type	Description
is_obs_created	N/A	Yes	Boolean	Whether a new OBS bucket can be created. • When the value is **true**, you can create an OBS bucket to store trace files. • When the value is **false**, you can select an existing OBS bucket to store trace files. The bucket name is a string of 3 to 64 characters and can only contain digits, letters, hyphens (-), and periods (.). Only one period and hyphen can be contained in the string.
is_support_trace_files_encryption	N/A	Yes	Boolean	Whether the function of encrypting stored trace files is enabled or not. This parameter must be used with **kms_id**.
kms_id	N/A	Yes	String	Key ID used for dumping and encrypting trace files. This key ID is obtained from KMS. This parameter is mandatory when the value of **is_support_trace_files_encryption** is **true**.
lts	is_lts_enabled	Yes	Boolean	Whether the LTS search function is enabled or not.
	log_group_name	Yes	String	Name of the log group that CTS creates in LTS.
	log_topic_name	Yes	String	Name of the log topic that CTS creates in LTS.
log_file_validate	N/A	No	json	Whether to enable the trace file verification function.
-	is_support_validate	Yes	Boolean	Whether the trace file verification function is supported or not.

- Example request

```
PUT /v1.0/{project_id}/tracker/system
{
    "bucket_name": "my_created_bucket",
    "file_prefix_name": "some_folder",
    "is_obs_created": true,
    "is_support_trace_files_encryption": true,
    "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",
    "lts": {
        "is_lts_enabled": true,
        "log_group_name": "CTS",
        "log_topic_name": "system-trace"
    },
    "log_file_validate": {
        "is_support_validate": true
    },
    "status": "disabled"
}
```

Response

- Parameters

Table 5-13 Parameters in the response

Parameter	Sub-Parameter	Type	Description
tracker_name	N/A	String	Tracker name. Currently, only tracker "system" is available.
bucket_name	N/A	String	OBS bucket name. Starts with a digit or letter and contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.)
file_prefix_name	N/A	String	Prefix of a log that needs to be stored in an OBS bucket.
status	N/A	String	Status of a tracker. The value can be **enabled** or **disabled**.

Parameter	Sub-Parameter	Type	Description
is_obs_created	N/A	Boolean	Whether a new OBS bucket can be created. ● When the value is **true**, you can create an OBS bucket to store trace files. ● When the value is **false**, you can select an existing OBS bucket to store trace files. The bucket name is a string of 3 to 64 characters and can only contain digits, letters, hyphens (-), and periods (.). Only one period and hyphen can be contained in the string.
is_support_trace_files_encryption	N/A	Boolean	Whether the function of encrypting stored trace files is enabled or not. This parameter must be used with **kms_id**.
kms_id	N/A	String	Key ID used for dumping and encrypting trace files. This key ID is obtained from KMS. This parameter is mandatory when the value of **is_support_trace_files_encryption** is **true**.
lts	is_lts_enabled	Boolean	Name of the log group that CTS creates in LTS.
	log_topic_name	String	Name of the log topic that CTS creates in LTS.

Parameter	Sub-Parameter	Type	Description
	log_group_id	String	ID of the log group that CTS creates in LTS.
	log_topic_id	String	ID of the log topic that CTS creates in LTS.
log_file_validate	N/A	JSON object	Whether to enable the trace file verification function.
is_support_validate	N/A	Boolean	Whether the trace file verification function is supported or not.

- Example response

```
{
    "bucket_name": "my_created_bucket",
    "tracker_name": "system",
    "file_prefix_name": "some_folder",
    "is_obs_created": true,
    "is_support_trace_files_encryption": true,
    "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504",
    "lts": {
        "is_lts_enabled": true,
        "log_group_name": "CTS",
        "log_topic_name": "system-trace",
        "log_group_id": "b6895f40-b897-11e7-8aa6-2866d488c81c",
        "log_topic_id": "b68d5281-b897-11e7-8aa7-2866d488c81c"
    },
    "log_file_validate": {
        "is_support_validate": true
    },
    "status": "disabled"
}
```

Returned Value

- Normal

Table 5-14 Return code for successful requests

Returned Value	Description
200	The request is successfully processed.

- Abnormal

Table 5-15 Return code for failed requests

Returned Value	Description
400	The server failed to process the request.

Returned Value	Description
404	The server failed to find the requested resource.
500	Failed to complete the request because of an internal service error.
401	Your access request is rejected.
403	You are forbidden to access the requested page.

5.2.3 Querying a Tracker

Function

This API is used to query a tracker created on the **Tracker Information** page. The detailed information includes the name of the tracker, name of the OBS bucket for storing traces, and prefix of the traces stored in an OBS bucket.

URI

GET /v1.0/{project_id}/tracker?tracker_name={tracker_name}

For details about the parameters, see **Table 1**.

Table 5-16 Parameters in the URI

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
tracker_name	No	String	Tracker name. If this parameter is not specified, all trackers will be queried. Currently, only tracker "system" is available.

NOTE

CTS may support multiple trackers. If the query request **GET /v1.0/{project_id}/tracker** does not contain parameter **tracker_name**, the response is in array format. Otherwise, the response is in object format.

Request

- Parameters

  None.

- Example request

```
GET /v1.0/{project_id}/tracker?tracker_name=system
```

Response

- Parameters

Table 5-17 Parameters in the request

Parameter	Sub-Parameter	Type	Description
tracker_name	N/A	String	Tracker name. Currently, only tracker "system" is available.
bucket_name	N/A	String	OBS bucket name. Starts with a digit or letter and contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.)
file_prefix_name	N/A	String	Prefix of a log that needs to be stored in an OBS bucket.
status	N/A	String	Status of a tracker. The value can be **enabled**, **disabled**, or **error**. If the value is set to **error**, the **detail** field is required for specifying the source of the error.

Parameter	Sub-Parameter	Type	Description
detail	N/A	String	Used to indicate the tracker exception cause. It is only returned in the response when the tracker status becomes abnormal. The value of this parameter can be one of the following: • **bucketPolicyError** • **noBucket** • **arrears**
is_obs_created	N/A	Boolean	Whether a new OBS bucket can be created. • When the value is **true**, you can create an OBS bucket to store trace files. • When the value is **false**, you can select an existing OBS bucket to store trace files. The bucket name is a string of 3 to 64 characters and can only contain digits, letters, hyphens (-), and periods (.). Only one period and hyphen can be contained in the string.
is_support_trace_files_encryption	N/A	Boolean	Whether the function of encrypting stored trace files is enabled or not. This parameter must be used with **kms_id**.

Parameter	Sub-Parameter	Type	Description
kms_id	N/A	String	Key ID used for dumping and encrypting trace files. This key ID is obtained from KMS. This parameter is mandatory when the value of **is_support_trace_files_encryption** is **true**.

- Example response

```
{
 "bucket_name" : "my_created_bucket",
 "tracker_name" : "system",
 "detail" : "noBucket",
 "file_prefix_name" : "some_folder",
 "status" : "disabled",
 "is_obs_created": true,
 "is_support_trace_files_encryption": true,
 "kms_id": "13a4207c-7abe-4b68-8510-16b84c3b5504"
}
```

Returned Value

- Normal

Table 5-18 Return code for successful requests

Returned Value	Description
200	The request is successful and the query result is returned.

- Abnormal

Table 5-19 Return code for failed requests

Returned Value	Description
400	The server failed to process the request.
500	Failed to complete the request because of an internal service error.
401	Your access request is rejected.
403	You are forbidden to access the requested page.

5.2.4 Deleting a Tracker

Function

This API is used to delete an existing tracker from the Cloud Trace Service (CTS) console. Deleting a tracker has no impact on the operation records generated. When you enable CTS again, you can view those operation records.

URI

DELETE /v1.0/{project_id}/tracker?tracker_name={tracker_name}

For details about the parameters, see **Table 1**.

Table 5-20 Parameters in the URI

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
tracker_name	No	String	Tracker name. If this parameter is not specified, the global tracker will not be deleted.

Request

- Parameters

  None.
- Example request

  ```
  DELETE /v1.0/{project_id}/tracker?tracker_name=system
  ```

Response

- Parameters

  None.
- Example response

  None.

Returned Values

- Normal

Table 5-21 Return code for successful requests

Returned Value	Description
204	Deleted successfully.

- Abnormal

Table 5-22 Return code for failed requests

Returned Value	Description
400	The server failed to process the request.
404	The server failed to find the requested resource or deleting some trackers failed.
500	The request failed to be executed or some trackers failed to be deleted.
401	Your access request is rejected.
403	You are forbidden to access the requested page.

5.3 Trace Management

5.3.1 Querying a Trace List

Function

This API is used to query records of operations on resources during the last seven days.

URI

GET /v2.0/{project_id}/{tracker_name}/trace{?trace_id,service_type,resource_type,resource_id,resource_name,trace_name,trace_status,user,limit,from,to,next}

For details about the parameters, see **Table 1**.

Table 5-23 Parameters in the URI

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
tracker_name	Yes	String	Currently, only tracker "system" is available.

Request

- Parameters

Table 5-24 Parameters in the request

Parameter	Mandatory	Type	Description
service_type	No	String	Type of a service whose traces are to be queried. The value must be the abbreviation of a cloud service that has been interconnected with CTS. It is a word composed of uppercase letters. For the interconnected cloud services, see section "Supported Services" in the *Cloud Trace Service User Guide*.
resource_type	No	String	Type of a resource whose traces are to be queried. The value is a string of 1 to 64 characters and can contain uppercase and lowercase letters (a to z and A to Z), digits (0 to 9), hyphens (-), underscores (_), and periods (.). In addition, it must start with a letter. For the interconnected cloud services, see section "Supported Services" in the *Cloud Trace Service User Guide*.
resource_id	No	String	ID of a resource whose traces are to be queried.
resource_name	No	String	Name of a resource whose traces are to be queried. **NOTE** The value may contain uppercase letters.
trace_name	No	String	Operation recorded by this trace. **NOTE** The value may contain uppercase letters.
limit	No	String	Number of traces returned in the trace list. The default value is **50** and the maximum value is **200**.

Parameter	Mandatory	Type	Description
next	No	String	Time of a queried trace (you can query traces earlier than the time). The value can be the parameter **marker** value in **Table 5**. The time condition **next** can be used with another time condition **from** and **to**. The final query condition is the intersection of the preceding two conditions.
from	No	String	Starting timestamp of the queried trace. The value is in UTC format, accurate to ms, and contains 13 digits. The time spent in selecting this filter is excluded. **from** and **to** are both used.
to	No	String	Ending timestamp of the queried trace. The value is in UTC format, accurate to ms, and contains 13 digits. The time spent in selecting this filter is excluded. The query criteria **to** and **from** are both used.
trace_id	No	String	ID of a trace. If this parameter is used as a filter, other filters cannot be selected.
trace_status	No	String	Status of a trace. The value can be **normal**, **warning**, or **incident**.
user	No	String	Specified username whose traces are to be queried. **NOTE** The value may contain uppercase letters.

- Example request

```
GET
/v2.0/{project_id}/{tracker_name}/trace?
limit=11&to=1479095278000&from=1478490478000&trace_name=createTracker&resource_type=tracker&service_type=CTS
```

Response

- Parameters

Table 5-25 Parameters in the response

Parameter	Type	Description
traces	array	Trace array in the trace list.
meta_data	Structure	Extended field. The value can be **count** (number of traces in the response) or **marker** (ID of the last trace in the trace list).

Table 5-26 Data structure description of the **traces** field

Parameter	Type	Description
resource_id	String	ID of a resource on which operations are performed.
trace_name	String	Name of a trace. The value is a string of 1 to 64 characters and can contain uppercase and lowercase letters (a to z and A to Z), digits (0 to 9), hyphens (-), underscores (_), and periods (.). In addition, it must start with a letter.
trace_status	String	Status of a trace. The value can be **normal**, **warning**, or **incident**.
trace_type	String	Resource of a trace. The value can be **ApiCall**, **ConsoleAction**, or **SystemAction**.
request	Structure	Request of an operation on resources.
response	Structure	Response to a user request, that is, the returned information for an operation on resources.
code	String	Records the response to a user request and specifies the HTTP status code returned by the API recorded in the trace.
api_version	String	Version of the API.
message	String	Remarks added by other cloud services to a trace.
record_time	Long	Timestamp when a trace is recorded by CTS.
trace_id	String	ID of a trace. The value is the UUID generated by the system.
time	Long	Time when a trace occurs.

Parameter	Type	Description
user	Structure	User information for which a trace is triggered.
service_type	String	Type of a service whose traces are to be queried. The value must be the abbreviation of a cloud service that has been interconnected with CTS. It is a word composed of uppercase letters.
resource_type	String	Type of a resource whose traces are to be queried. The value is a string of 1 to 64 characters and can contain uppercase and lowercase letters (a to z and A to Z), digits (0 to 9), hyphens (-), underscores (_), and periods (.). In addition, it must start with a letter.
source_ip	String	IP address of a user for whom a trace is triggered.
resource_name	String	Name of a resource whose traces are to be queried.
request_id	String	Record the ID of the request.
location_info	String	Additional information required for fault locating after a request recording error occurs.
endpoint	String	This operation involves the endpoint of the page that displays cloud resource details.
resource_url	String	This operation involves the access link (excluding the endpoint) of the page that displays cloud resource details.

Table 5-27 Data structure description of the **meta_data** field

Parameter	Type	Description
count	Integer	Number of traces returned in the trace list.
marker	String	ID of the last trace in the trace list. The value of this parameter can be used as the **next** value. If the value of **marker** is **null**, all traces have been returned.

- Example response

```
{
  "traces" : [ {
```

```
    "time" : 1472148708232,
    "user" : {"name":"xxx","id":"a2e899190fcd444084a68fc0ac2sc1e9","domain":
{"name":"xxx","id":"05b2598d69bc4a209f9ac5eeeb1f91ad"}},
    "response" : {"code":"VPC.0514","message":"Update port fail."},
    "code" : 200,
    "service_type" : "VPC",
    "resource_type" : "eip",
    "resource_name" : "192.144.163.1",
    "resource_id" : "d502809d-0d1d-41ce-9690-784282142ccc",
    "trace_name" : "deleteEip",
    "trace_status " : "warning",
    "trace_type" : "ConsoleAction",
    "api_version" : "2.0",
    "record_time" : 1481066128032,
    "trace_id" : "e001ccb9-bc09-11e6-b00b-4b2a61338db6" ,
    "request_id" : "a0001c1b9-bctt-2136-c12b-4b2a611116",
    "location_info" : "resource has been deleted",
    "endpoint" : "https://*****/vpc?agencyId=***&region=***&locale=zh-cn#",
    "resource_url" : "/vpc/vpcmanager/vpcs?vpcid=*****"
  }, {
    "time" : 1472148708232,
    "user" : {"name":"xxx","id":"a2e899190fcd444084a68fc0ac2sc1e9","domain":
{"name":"xxx","id":"05b2598d69bc4a209f9ac5eeeb1f91ad"}},
"request": {"servers":[{"id":"3045f042-9a7c-436d-a944-
ff76ceb7b477"}],"delete_volume":false,"delete_publicip":false},
    "response" : {"code":"VPC.0514","message":"Update port fail."},
    "code" : 200,
    "service_type" : "VPC",
    "resource_type" : "eip",
    "resource_name" : "192.144.163.1",
    "resource_id" : "d502809d-0d1d-41ce-9690-784282142ccc",
    "trace_name" : "deleteEip",
    "trace_status" : "warning",
    "trace_type" : "ConsoleAction",
    "api_version" : "2.0",
    "record_time" : 1481066128032,
    "trace_id" : "e001ccb8-bc09-11e6-b2cc-2640a43cc6e8" ,
    "request_id" : "a0001c1b9-bctt-2136-c12b-4b2a611116",
    "location_info" : "resource has been deleted",
    "endpoint" : "https://*****/vpc?agencyId=***&region=***&locale=zh-cn#",
    "resource_url" : "/vpc/vpcmanager/vpcs?vpcid=*****"

  } ],
  "meta_data" : {
    "count" : 2,
    "marker" : "e001ccb8-bc09-11e6-b2cc-2640a43cc6e8"
  }
}
```


Returned Value

- Normal

  Table 5-28 Return code for successful requests

Returned Value	Description
200	The request is successful and the query result is returned.

- Abnormal

Table 5-29 Return code for failed requests

Returned Value	Description
400	The query parameters are abnormal.
500	Failed to complete the request because of an internal service error.
401	Your access request is rejected.
403	You are forbidden to access the requested page.
404	The requested trace does not exist.

5.3.2 Querying Global Traces

Function

This API is used to query global traces reported by cloud services, such as login and logout traces reported by IAM.

This API can be called only by the domain name of the cts.cn-north-4.myhuaweicloud.com in North China – Beijing4.

URI

GET /v2.0/domains/{domainId}/traces?pageIndex,pageSize,event_type,type,isDesc,endTime,startTime,service_type,user,trace_rating,resource_id,record_time

For details about the parameters, see **Table 1**.

Table 5-30 Parameters in the URI

Parameter	Mandatory	Type	Description
domainId	Yes	String	Tenant ID.
pageIndex	No	int	Page
event_type	Yes	String	Trace source. Currently, the value is fixed to **global**.
resource_type	No	String	Resource type
type	No	String	Query mode. When the query mode is **next** or **pre**, **record_time** must be used. • "init": First query • "next": Downward query • "pre": Upward query

Parameter	Mandatory	Type	Description
isDesc	No	Boolean	Ascending or descending order
startTime	No	long	Query start time.
endTime	No	long	Query end time.
service_type	No	String	Service name
user	No	String	User name
trace_rating	No	String	Event Status
resource_id	No	String	Resource ID.
record_time	No	long	Record time (used for sorting and similar to lineNum). It is used when a query is performed downward or upward. Its value is the value of record_time in the last record.

Request

- Example request

```
GET
Initial request
/v2.0/domains/2306579dc99f4c8690b14b68e734fcd9/traces?
pageIndex=1&pageSize=50&event_type=global&type=init&isDesc=true&isConsole=true&endTime=1567
479037112&startTime=1566874237112&service_type=IAM&user=paas_cts_z00418070_01&resource_ty
pe=user&trace_rating=normal

Downward query
/v2.0/domains/2306579dc99f4c8690b14b68e734fcd9/traces?
pageIndex=1&pageSize=50&event_type=global&type=next&isDesc=true&isConsole=true&record_time=
1567071858959000001&endTime=1567479122688&startTime=1566874322688&service_type=IAM&use
r=paas_cts_z00418070_01&resource_type=user&trace_rating=normal
```

Response

- Parameter description

Table 5-31 Parameters in the response

Parameter	Type	Description
total	int	Total number of the queried traces
data	array	Queried traces

Table 5-32 Data structure description of the **data** field

Parameter	Type	Description
resource_id	String	ID of the cloud service resource corresponding to the trace.
trace_name	String	Name of the trace in the trace list.
trace_status	String	Trace status. The options are as follows: normal, warning and incident.
trace_type	String	Trace type, including API calling (ApiCall), console page calling (ConsoleAction), and inter-system calling (SystemAction).
request	Structure	Request of an operation on resources.
response	Structure	Response to the user request, that is, the response to the resource operation result.
code	String	The HTTP status code returned by the API recorded in the trace.
api_version	String	Version of the cloud service API.
message	String	Remarks added by other cloud services to the trace.
record_time	Long	Trace time recorded by CTS.
trace_id	String	Trace ID, which is automatically generated by the system.
time	Long	Time when a trace occurs.
user	Structure	Information about the user who triggers the trace.
service_type	String	Type of a cloud service whose traces are to be queried. The value must be the abbreviation of a cloud service that has been interconnected with CTS. It is a word composed of uppercase letters.
resource_type	String	Type of a resource whose traces are to be queried.
source_ip	String	IP address of the tenant that triggers the trace.
resource_name	String	Name of the resource corresponding to the trace.

Parameter	Type	Description
request_id	String	Request ID.
location_info	String	Auxiliary information required for locating a request error.
endpoint	String	Cloud resource details page
resource_url	String	Link to the cloud resource details page (excluding the endpoint)

- Example response

{"total":50,"data":[{"context":"{\"code\":\"302\",\"source_ip\":\"100.79.4.140\",\"trace_type\":\"ConsoleAction\",\"event_type\":\"<HighLightTag>global</HighLightTag>\",\"project_id\":\"16edf66e79d04187acb99a463e610764\",\"trace_id\":\"f95d3c01-cdf1-11e9-8b03-011bbe3a6405\",\"trace_name\":\"login\",\"resource_type\":\"<HighLightTag>user</HighLightTag>\",\"trace_rating\":\"<HighLightTag>normal</HighLightTag>\",\"service_type\":\"<HighLightTag>IAM</HighLightTag>\",\"resource_id\":\"f3f18b9215014f0d9ded3045af020811\",\"tracker_name\":\"<HighLightTag>global</HighLightTag>\",\"time\":\"1567477476534\",\"resource_name\":\"<HighLightTag>paas_cts_z00418070_01</HighLightTag>\",\"record_time\":\"1567477476596\",\"<HighLightTag>user</HighLightTag>\":\"{\\\"name\\\":\\\"<HighLightTag>paas_cts_z00418070_01</HighLightTag>\\\",\\\"id\\\":\\\"f3f18b9215014f0d9ded3045af020811\\\",\\\"domain\\\":{\\\"name\\\":\\\"<HighLightTag>paas_cts_z00418070_01</HighLightTag>\\\",\\\"id\\\":\\\"2306579dc99f4c8690b14b68e734fcd9\\\"}}\"}","record_time":"1567477476596000001"}]}

Returned Value

- Normal

Table 5-33 Return code for successful requests

Returned Value	Description
200	The request is successful and the query result is returned.

- Abnormal

Table 5-34 Return code for failed requests

Returned Value	Description
403	Unauthorized.
500	Failed to complete the request because of an internal service error.
404	The requested trace does not exist.

6 Appendix

6.1 Error Code

Description

This section explains the meanings of error code returned by CTS APIs.

Example of Returned Error Information

{"details":{"details":"Create bucket failed","code":"cts.0036","obs_details":
{"obs_code":403,
"obs_error":"ClusterGroupExclusiveException","obs_message":"The cluster groups are exclusively,you are not allowed to create bucket."}}}

NOTE

The **obs_details** field is only contained in the error message returned by the OBS bucket.

Error Code Description

Table 6-1 Error code description

Response Code	Error Code	Description	Error Message	Details
400	cts.0001	The API version number is empty.	Empty API version. Check whether you need to specify an API version.	The API version in the URL is empty. Check whether the API version information is specified.

Response Code	Error Code	Description	Error Message	Details
400	cts.0002	The API version number is incorrect.	Invalid API version. Check whether the API version is correct.	The API version information in the URL is invalid. Check whether the API version information is correct.
400	cts.0003	The tenant ID does not exist.	Empty project ID. Check whether you need to specify a project ID.	The project ID in the URL is empty. Check whether the project ID is specified.
400	cts.0004	The tenant ID is incorrect.	Invalid project ID. Check whether the project ID is correct.	The project ID in the URL is invalid. Check whether the project ID is correct.
400	cts.0005	The query parameters are incorrect.	Invalid query criteria. Check whether query criteria parameters are correct.	Invalid query criteria. Check whether query criteria parameters are correct. **** does not exist.
400	cts.0006	The request URL does not exist.	The URL does not exist. Check whether the URL meets requirements.	The URL does not exist. Check whether the URL is correct.
400	cts.0007	Request parameters are incorrect.	Invalid message body. Different errors are returned according to different parameters.	Invalid message body. Check whether the message body format and content are correct. Invalid message body. The message body is not needed or not JSON.
500	cts.0008	Data reading error.	Data read exception. Contact O&M personnel.	Data reading is abnormal. Contact O&M personnel.
500	cts.0009	Data writing error.	Data write exception. Contact O&M personnel.	Data writing is abnormal. Contact O&M personnel.
403	cts.0010	Repeated creation.	Tracker is existed already.	The tracker already exists. Modify the tracker name and create it again.

Response Code	Error Code	Description	Error Message	Details
403	cts.0011	The user does not have the permission to perform the operation.	The user fails the authentication or does not have permission to this operation.	The authentication failed or you have no rights to perform the operation.
404	cts.0012	The tracker does not exist.	The tracker does not exist. Check whether the tracker name is correct or CTS has been enabled.	The tracker does not exist. Check whether the tracker name is correct.
404	cts.0013	The trace does not exist.	The event does not exist. Check whether the trace ID is correct.	The trace does not exist. Check whether the trace ID is correct.
400	cts.0014	The token is not found.	No valid token found.	The token is not found. Contact O&M personnel.
500	cts.0015	Internal service error.	Internal server error. Contact the O&M personnel.	An internal service error occurs. Contact O&M personnel.
500	cts.0016	Failed to write data into the cache.	Failed to write data into the cache. Contact O&M personnel.	Cache writing failed. Contact O&M personnel.
401	cts.0017	Token authentication failed.	The token does not exist or is invalid. Check whether the token information is correct.	The token does not exist or is invalid. Check whether the token information is correct.
202	cts.0018	O&M query is not supported.	The current environment does not support the calling of O&M interfaces. Check whether the URL is correct.	The system does not support invocation of O&M APIs. Check whether the URL information is correct.
400	cts.0019	Failed to obtain the AK/SK.	Failed to obtain the user AK and SK.	Failed to obtain the AK/SK of the current user. Contact O&M personnel.
400	cts.0020	Invalid AK/SK.	The user AK and SK are invalid.	The AK/SK of the current user is invalid.

Response Code	Error Code	Description	Error Message	Details
400	cts.0021	OBS authentication failed.	Failed to authenticate the OBS bucket. Contact O&M personnel.	The OBS bucket does not exist. Check whether the OBS bucket has been deleted.
400	cts.0022	Enabled.	CTS cannot be repeatedly enabled. Check whether CTS has been enabled.	The OBS bucket authentication failed. Contact O&M personnel.
404	cts.0023	The OBS bucket has been deleted.	The OBS bucket of the current user does not exist. Check whether the OBS bucket has been deleted.	The requested OBS bucket does not exist. Check whether the OBS bucket has been deleted.
403	cts.0024	OBS authentication failed.	CTS is not authorized by the current user's OBS bucket. Check whether CTS is successfully authorized.	CTS failed to obtain the permission to access the OBS bucket. Check whether CTS has been authorized.
400	cts.0025	Connecting to the database failed.	The api service disconnect with db.	The API service failed to connect to the database.
400	cts.0026	Connecting to the cache failed.	The api service disconnect with cache.	The API service failed to connect to the Cache node.
400	cts.0027	Failed to obtain the bucket list.	Failed to obtain the OBS bucket list. Contact O&M personnel.	Failed to obtain the list of OBS buckets. Contact O&M personnel.
400	cts.0028	Failed to obtain the bucket location.	Failed to obtain the OBS bucket location. Contact O&M personnel.	Failed to obtain the region of the OBS bucket. Contact O&M personnel.
400	cts.0029	Failed to create a bucket policy.	Failed to create the OBS bucket policy. Contact O&M personnel.	Failed to create an OBS bucket policy. Contact O&M personnel.

Response Code	Error Code	Description	Error Message	Details
400	cts.0030	Failed to delete a bucket policy.	Failed to delete the OBS bucket policy. Contact O&M personnel.	Failed to delete an OBS bucket policy. Contact O&M personnel.
400	cts.0032	Failed to authenticate the account information.	Failed to check the OBS bucket existence. Contact O&M personnel.	The OBS account information is not updated, and you can try 15 minutes later.
400	cts.0033	The OBS bucket does not belong to the current user.	Account information is not refresh in OBS. Please retry after 15 minutes.	The OBS bucket does not belong to the current user. Check whether the OBS bucket is correct.
400	cts.0034	Failed to obtain the central region from IAM.	The OBS bucket does not belong to the current user. Check whether the OBS bucket is correct.	Failed to obtain the central area information from IAM.
400	cts.0036	Failed to create a bucket.	Create bucket failed.	Failed to create a bucket.
400	cts.0037	Failed to check the bucket.	Check bucket failed.The bucket is already exist.	Failed to check the bucket, and a bucket exists.
400	cts.0038	Failed to obtain the project ID.	Failed to obtain the project ID from IAM when the system is trying to obtain keys from KMS.	Failed to obtain the project ID. Contact O&M personnel.
400	cts.0039	The key does not exist.	The KMS key id of the current user does not exist. Check whether the KMS key id has been deleted.	The key does not exist. Contact O&M personnel.
400	cts.0040	Failed to obtain the key.	Failed to obtain key list from KMS.	Failed to obtain the key. Contact O&M personnel.

Response Code	Error Code	Description	Error Message	Details
400	cts.0041	Failed to create a log topic or group.	Create log group or log topic failed, create tracker failed, please try again.	Failed to create a log topic or group. Contact O&M personnel.
400	cts.0043	Failed to query the bucket.	Check bucket failed. Contact O&M personnel.	Failed to query the bucket. Contact O&M personnel.
400	cts.0044	Failed to set the bucket lifecycle.	Set bucket life cycle failed. Contact O&M personnel.	Failed to set the bucket lifecycle. Contact O&M personnel.
400	cts.0045	Failed to obtain the bucket lifecycle.	Get bucket life cycle failed. Contact O&M personnel.	Failed to get the bucket lifecycle. Contact O&M personnel.
400	cts.0046	The number of created trackers reaches the upper limit.	The maximum number of trackers is: %s.	Failed to create the tracker. Contact O&M personnel.
400	cts.0048	Failed to delete the data tracker.	Failed to delete the tracker, Please try again.	Failed to delete the tracker. Contact O&M personnel.
400	cts.0049	A bucket cannot be used to create multiple trackers that record operations of the same type.	You cannot create different trackers to record the same type of operations on the same OBS bucket.	Failed to create the tracker. Contact O&M personnel.

Response Code	Error Code	Description	Error Message	Details
400	cts.0050	The tracked bucket cannot be modified.	The tracked OBS bucket can not be modify.	You are not allowed to modify the tracked OBS bucket.
400	cts.0051	A bucket cannot be tracked by multiple trackers.	Failed to configure the tracker because the new settings could not be applied to the log group or log topic. Try again later.	An OBS bucket can be tracked by only one tracker.
400	cts.0052	Failed to delete the log topic.	The topic in LTS deleted failed, tracker deleted failed. Please try again or delete the topic manually.	Failed to delete the log topic. Contact O&M personnel.
400	cts.0054	The key operation notification already exists.	Notification name is existed already.	The key operation notification already exists and cannot be created repeatedly.
404	cts.0055	The key operation notification does not exist.	The notification does not exist.	The key operation notification does not exist. Contact O&M personnel.
400	cts.0056	The number of key operation notifications reaches the upper limit.	The quantity has exceeded the maximum quantity limit.	Failed to create the key operation notification. Contact O&M personnel.
400	cts.0057	Failed to start or stop the key operation notification.	State setting failed, check if topicId exists.	Failed to start or stop the key operation notification. Contact O&M personnel.

Response Code	Error Code	Description	Error Message	Details
400	cts.0058	Failed to obtain group users from IAM.	Get group users failed from IAM.	Failed to obtain group users from IAM. Contact O&M personnel.
400	cts.0059	An error occurred when querying the **page_type** parameter in the CDR.	Get meter data failed, the page_type parameter must be 2.	The **page_type** parameter must be set to **2**.
400	cts.0060	An error occurred when querying the **limit** parameter in the CDR.	Get meter data failed, the limit parameter should be greater than 0 or equal to 1000.	The **limit** parameter must be set to **0-1000**.
401	cts.0061	The project ID in the request does not match that in the token.	CSB update CTS status error, projectId in token is not equal projectId in body.	The project ID in the token is different from that in the request. Contact O&M personnel.
404	cts.0062	The tracked bucket has been deleted.	The OBS bucket of the current user does not exist. Check whether the tracked bucket has been deleted.	Check whether the tracked bucket has been deleted.
404	cts.0063	The current version number cannot be queried.	CTS does not support API interface version query.	The current version number cannot be queried.

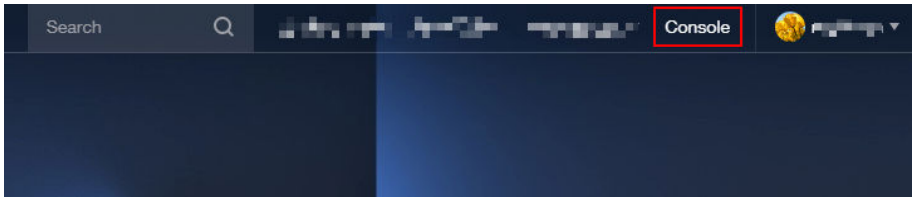
Response Code	Error Code	Description	Error Message	Details
400	cts.0065	Failed to get time zone and language from CBC.	Failed to get time zone and language from CBC.	Failed to get time zone and language from CBC. Contact O&M personnel.
500	cts.0066	Failed to obtain the AK/SK from IAM.	Failed to get AK/SK from IAM.	Failed to obtain the AK/SK from IAM. Contact O&M personnel.

6.2 Obtaining the Account ID and Project ID

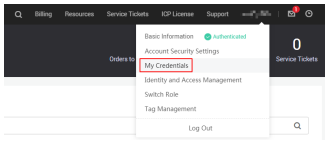
Obtaining Account and Project IDs from the Console

Account and project IDs are required for some URLs when an API is called. You can perform the following operations to obtain the IDs:

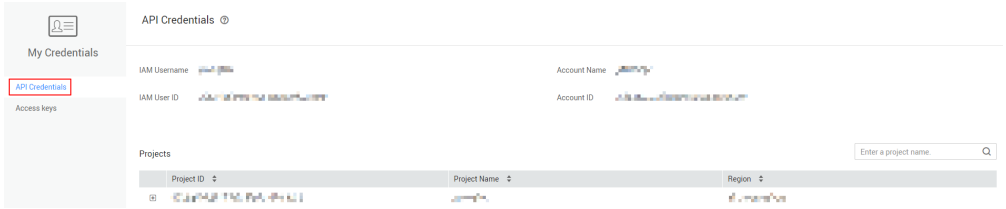
1. Log in to the management console.



2. Click the username and select **My Credentials** from the drop-down list.



3. On the **My Credentials** page, view the account and project IDs.



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

Obtaining the Project ID by Calling an API

The API for obtaining the project ID is **GET https://{Endpoint}/v3/projects/**. **{Endpoint}** indicates the endpoint of IAM (Identity and Access Management).

The following is an example response. The value of **id** in the **projects** field is the project ID.

```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "cn-north-4",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```