

Data Security Center

API Reference

Issue 02
Date 2024-03-30



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

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

| | |
|--------------------------------------------------------------------------------|-----------|
| 1 Before You Start..... | 1 |
| 1.1 Overview..... | 1 |
| 1.2 API Calling..... | 1 |
| 1.3 Endpoints..... | 1 |
| 1.4 Limitations and Constraints..... | 2 |
| 1.5 Basic Concepts..... | 2 |
| 1.6 Selecting an API Type..... | 3 |
| 2 Calling APIs..... | 4 |
| 2.1 Making an API Request..... | 4 |
| 2.2 Authentication..... | 6 |
| 2.3 Response..... | 8 |
| 3 API Description..... | 10 |
| 3.1 Data Watermarking..... | 10 |
| 3.1.1 Injecting a Data Watermark..... | 10 |
| 3.1.2 Extracting a Data Watermark..... | 13 |
| 3.2 Resource Management..... | 16 |
| 3.2.1 Instance Ordering..... | 16 |
| 3.2.2 Querying Resource Provisioning Information..... | 19 |
| 3.3 Dynamic Data Masking..... | 22 |
| 3.3.1 Masking Sensitive Data..... | 22 |
| 3.4 Alarm Notifications..... | 26 |
| 3.4.1 Querying an Alarm Notification Topic..... | 26 |
| 3.4.2 Modifying an Alarm Notification Topic..... | 28 |
| 3.5 Image Watermarking..... | 30 |
| 3.5.1 Injecting Invisible Watermarks into Images..... | 30 |
| 3.5.2 Extracting Invisible Text Watermarks..... | 32 |
| 3.5.3 Extracting Invisible Watermarks from Images..... | 34 |
| 3.5.4 Injecting Invisible Watermarks into Images (Image Addresses)..... | 36 |
| 3.5.5 Extracting Dark Watermarks from Images (Image Addresses)..... | 39 |
| 3.5.6 Extracting Invisible Image Watermarks from Images (Image Addresses)..... | 42 |
| 3.6 Asset Management..... | 45 |
| 3.6.1 Editing a Data Asset Name..... | 45 |

| | |
|--------------------------------------------------------------------------------|------------|
| 3.6.2 Querying the Data Asset Scanning Authorization..... | 47 |
| 3.6.3 Adding a Data Asset Scanning Authorization..... | 49 |
| 3.6.4 Deleting a Data Asset Scanning Authorization..... | 51 |
| 3.7 Document Watermarking..... | 53 |
| 3.7.1 Injecting Watermarks into Documents..... | 53 |
| 3.7.2 Extracting Invisible Watermarks..... | 57 |
| 3.7.3 Injecting Watermarks into Documents (Document Addresses)..... | 59 |
| 3.7.4 Extracting Invisible Watermarks from Documents (Document Addresses)..... | 64 |
| 3.8 Sensitive Data Discovery..... | 66 |
| 3.8.1 Querying the Identification Task List..... | 67 |
| 3.8.2 Querying the Result of an Identification Task..... | 69 |
| 3.8.3 Viewing the Rule List..... | 74 |
| 3.8.4 Creating a Sensitive Data Scanning Rule..... | 77 |
| 3.8.5 Modifying a Sensitive Data Scanning Rule..... | 80 |
| 3.8.6 Deleting a Sensitive Data Scanning Rule..... | 82 |
| 3.8.7 Querying Sensitive Data Scanning Rule Groups..... | 84 |
| 3.8.8 Creating a Sensitive Data Scanning Rule Group..... | 86 |
| 3.8.9 Deleting a Sensitive Data Scanning Rule Group..... | 88 |
| 3.8.10 Creating a Sensitive Data Scanning Task..... | 90 |
| 3.8.11 Deleting a Sensitive Data Scanning Task..... | 92 |
| 3.9 Static Data Masking..... | 94 |
| 3.9.1 Querying the Data Masking Task Execution List..... | 94 |
| 3.9.2 Starting or Stopping a Data Masking Task..... | 97 |
| 3.10 API Call Records..... | 98 |
| 3.10.1 Querying OpenAPI Calls..... | 98 |
| A Appendixes..... | 102 |
| A.1 Status Codes..... | 102 |
| A.2 Error Codes..... | 103 |
| A.3 Obtaining a Project ID..... | 106 |
| A.4 Configuring a Dynamic Sensitive Data Masking Policy..... | 107 |
| A.4.1 SHA-256/512..... | 107 |
| A.4.2 AES..... | 108 |
| A.4.3 PRESNM..... | 109 |
| A.4.4 MASKNM..... | 111 |
| A.4.5 PRESXY..... | 112 |
| A.4.6 MASKXY..... | 114 |
| A.4.7 SYMBOL..... | 116 |
| A.4.8 KEYWORD..... | 118 |
| A.4.9 NUMERIC..... | 118 |
| B Change History..... | 120 |

1 Before You Start

1.1 Overview

Data Security Center (DSC) is a latest-generation cloud data security management platform that protects your assets with functions such as risk classification, sensitive data identification, watermark source tracing, and static data masking. DSC monitors data security and gives you a comprehensive view of your data security on the cloud through Asset Map.

This document describes how to use the APIs provided by DSC to perform operations such as database watermarking, image watermarking, and data masking.

1.2 API Calling

DSC supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For more information, see [Calling APIs](#).

1.3 Endpoints

The following table describes the endpoint of DSC.

Table 1-1 DSC endpoint

| Region Name | Region | Endpoint | Protocol |
|---------------|----------------|--------------------------------------|----------|
| CN-Hong Kong | ap-southeast-1 | dsc.ap-southeast-1.myhuaweicloud.com | HTTPS |
| LA-Sao Paulo1 | sa-brazil-1 | sdg.sa-brazil-1.myhuaweicloud.com | HTTPS |

| Region Name | Region | Endpoint | Protocol |
|-------------|----------------|--------------------------------------|----------|
| AP-Bangkok | ap-southeast-2 | sdg.ap-southeast-2.myhuaweicloud.com | HTTPS |

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. Obtain the regions and endpoints from the enterprise administrator.

1.4 Limitations and Constraints

- [Table 1-2](#) lists the document and image watermarks supported by the DSC.

Table 1-2 Supported document/image watermark types

| Document/Image Watermark Carrier | Office (Windows and macOS) | WPS (Windows, macOS, Linux, and mobile OSs) | Adobe Reader | Chrome and Edge | Foxit PDF |
|----------------------------------|----------------------------|---------------------------------------------|--------------|-----------------|-----------|
| PDF | - | √ | √ | √ | √ |
| WORD | √ | √ | - | - | - |
| EXCEL | √ | √ | - | - | - |
| PPT | √ | √ | - | - | - |

NOTE

The symbol √ indicates that the carrier is supported by software.

- DSC APIs do not support inserting or extracting watermarks for OBS data. To perform watermark-related operations on OBS bucket data, store the data to the local PC and then call DSC APIs. The watermarked document will be returned in the response body.
- For more details of API constraints, see the API Description section.

1.5 Basic Concepts

- Account

An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be directly used to perform routine management. For security

purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- 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

Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified as universal regions and dedicated regions. A universal region provides universal cloud services for common users. A dedicated region provides services of the same type only or for specific users.

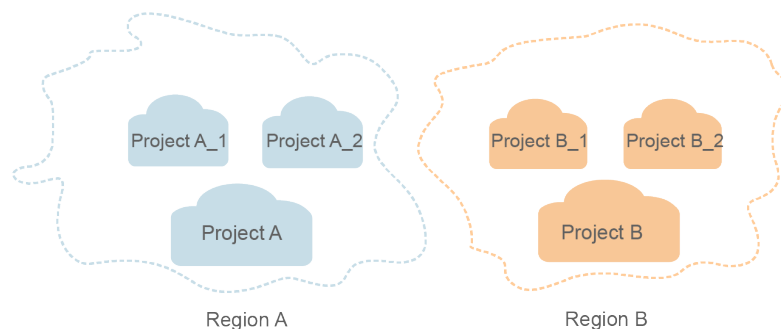
- Availability Zone (AZ)

An AZ comprises one or multiple physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Compute, 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 highly available systems across AZs.

- Project

Projects group and isolate 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. For more refined access control, create subprojects under a project and create resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

Figure 1-1 Project isolation model



1.6 Selecting an API Type

DSC provides V1 APIs.

2 Calling APIs

2.1 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 the administrator.
- **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 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 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.ap-southeast-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. The user token is a response to the API used to obtain a user token. This API is the only one that 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.

For more information, see [AK/SK-based 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.ap-southeast-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.

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. Set **username** to the name of a user, **domainname** to the name of the account that the user belongs to, ********* to the user's login password, and **xxxxxxxxxxxxxxxxxxxx** to the project name. You can learn more information about projects from the administrator.

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".

```
POST https://iam.ap-southeast-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": "xxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

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.

2.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- 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. This method is recommended because it provides higher security 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 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 token can be obtained by calling the required API. For more information, see Obtaining a User Token. A project-level token is required for calling this API, that is, **auth.scope** must be set to **project** in the request body. Example:

```
{
  "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.ap-southeast-1.myhuaweicloud.com/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, 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.

2.3 Response

Status Codes

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

A status code is a group of numbers ranging from 1xx to 5xx. It indicates the status of a response. For more information, see [Status Code](#).

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

Response Header

A response header corresponds to a request header, for example, **Content-Type**.

Figure 2-1 shows the response header for the API of , in which **x-subject-token** is the desired user token. Then, you can use the token to authenticate the calling of other APIs.

Figure 2-1 Header 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 → MIiYXQYJKoZIhvcNAQcCoIIYjCCGEOCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0B8wGgghacBIIWmHsidG9rZW4iOansiZXhwaXJlc19hdCI6IjpwMTktMDItMTNUMD
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkqjACgkqlqO1wi4JIGzrpd18LGXK5bdfq4lqHCYb8P4NaVONYejeAgzJVeFYtLWT1GSO0zxKZmlQHqJ82HBqHdglZO9fuEebL5dMhdavj+33wEI
xHRCE9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jggIFkNPQuFSOU8+uSsttVwrRtnfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUUVhVpxk8pxiX1wTEboX-
RzT6MUUbpvGw-oPNFYxJECKnoH3Hrozv0vN--n5d6Nbxg==
x-xss-protection → 1; mode=block;
```

(Optional) Response Body

A response body is generally returned in a structured format, corresponding to the **Content-Type** in the response header, and is used to transfer content other than the response header.

The following shows part of the response body for the API to . For the sake of space, only part of the content is displayed here.

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

```

If an error occurs during API calling, the system returns an error code and a message to you. The following shows the format of an error response body:

```
{
  "error": {
    "message": "The request you have made requires authentication.",
    "title": "Unauthorized"
  }
}
```

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

3 API Description

3.1 Data Watermarking

3.1.1 Injecting a Data Watermark

Function

This API is used to dynamically inject a watermark into the JSON body.

URI

POST /v1/{project_id}/sdg/database/watermark/embed

Table 3-1 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-2 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-3 Request body parameters

| Parameter | Mandatory | Type | Description |
|-------------------|-----------|--------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| watermark_content | Yes | String | Watermark content |
| watermark_key | Yes | String | Watermark key |
| columns | Yes | Array of Columns objects | List of field types. The number of field types in a list cannot exceed 100. At least two fields must be included. primary_key set to true indicates the primary key and primary_key set to false indicates watermark injection. |
| data | Yes | Array of Map<String, Object> objects | Content of a field, which can contain a maximum of 2000 characters. |

Table 3-4 Columns

| Parameter | Mandatory | Type | Description |
|-------------|-----------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| name | Yes | String | Field name, which can contain a maximum of 256 characters. |
| type | Yes | String | Field type. Enumeration values: <ul style="list-style-type: none">• INTEGER• STRING• DOUBLE |
| primary_key | Yes | Boolean | Whether a field in the database table is a primary key. |

Response Parameters

Status code: 200

Table 3-5 Response body parameters

| Parameter | Type | Description |
|-------------|--------------------------------------|------------------|
| marked_data | Array of Map<String, Object> objects | Watermarked data |

Status code: 400

Table 3-6 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Embed the watermark test12345678test. The watermark key is "keyword". The data field of the data type table is "item1". This field is the primary key.

POST https://{endpoint}/v1/{project_id}/sdg/database/watermark/embed

```
{
  "watermark_content": "test12345678test",
  "watermark_key": "keyword",
  "columns": [ {
    "name": "item1",
    "type": "INTEGER",
    "primary_key": true
  }, {
    "name": "item2",
    "type": "INTEGER",
    "primary_key": false
  } ],
  "data": [ {
    "item1": 0,
    "item2": 3
  }, {
    "item1": 1,
    "item2": 4
  } ]
}
```

Example Responses

Status code: 200

Request successful.

```
{
  "marked_data": [ {
    "item2": 3,
    "item1": "test"
  }, {
    "item2": 5,
```



```
"item1" : "test"  
}]  
}
```

Status code: 400

Parameter error.

```
{  
  "error_code" : "DSC.00000004",  
  "error_msg" : "Invalid parameter"  
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Parameter error. |

Error Codes

See [Error Codes](#).

3.1.2 Extracting a Data Watermark

Function

This API is used to extract a data watermark.

URI

POST /v1/{project_id}/sdg/database/watermark/extract

Table 3-7 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-8 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-9 Request body parameters

| Parameter | Mandatory | Type | Description |
|---------------|-----------|--------------------------------------|-----------------------------------------------------|
| watermark_key | Yes | String | Watermark key |
| columns | Yes | Array of Columns objects | Field type list, which cannot exceed 100. |
| data | Yes | Array of Map<String, Object> objects | Watermark data records, which cannot exceed 30,000. |

Table 3-10 Columns

| Parameter | Mandatory | Type | Description |
|-------------|-----------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| name | Yes | String | Field name, which can contain a maximum of 256 characters. |
| type | Yes | String | Field type. Enumeration values: <ul style="list-style-type: none">• INTEGER• STRING• DOUBLE |
| primary_key | Yes | Boolean | Whether a field in the database table is a primary key. |

Response Parameters

Status code: 200

Table 3-11 Response body parameters

| Parameter | Type | Description |
|------------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| watermarks | Array of strings | List of watermarks extracted. The uploaded data may include different watermarks. All watermarks extracted are returned, and the number of watermarks cannot exceed 100. |

Status code: 400**Table 3-12** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract data watermarks. The watermark key is "key", and the data field of the data type table is "col". This field is not a primary key.

```
POST https://{endpoint}/v1/{project_id}/sdg/database/watermark/extract
{
  "watermark_key": "key",
  "columns": {
    "name": "col",
    "type": "INTEGER",
    "primary_key": false
  },
  "data": {
    "col": 0.1
  }
}
```

Example Responses

Status code: 200

Request successful.

```
{
  "watermarks": [ "watermark" ]
}
```

Status code: 400

Parameter error.

```
{
  "error_code": "DSC.00000004",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Parameter error. |

Error Codes

See [Error Codes](#).

3.2 Resource Management

3.2.1 Instance Ordering

Function

This API is used to place an order based on the billing mode and billing period.

URI

POST /v1/{project_id}/period/order

Table 3-13 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-14 Request body parameters

| Parameter | Mandatory | Type | Description |
|----------------------|-----------|---------|------------------------------------------------------------------------------------------|
| charging_mode | Yes | Integer | Billing mode. The options are as follows: 0: yearly/monthly; 1: pay-per-use; 2: one-off. |
| cloud_service_type | Yes | String | Cloud service type |
| composite_product_id | No | String | Package ID |
| discount_id | No | String | Discount ID |

| Parameter | Mandatory | Type | Description |
|-----------------------|-----------|-----------------------------------------|--------------------------------------------------------------------------|
| is_auto_renew | No | Integer | Auto-renewal |
| period_num | Yes | Integer | Subscription periods |
| period_type | Yes | Integer | Subscription period type. The options are as follows: 2: month; 3: year. |
| product_infos | Yes | Array of ProductInfoBean objects | Product information list |
| promotion_activity_id | No | String | Promotion ID |
| promotion_info | No | String | Promotion information |
| region_id | Yes | String | ID of the region where the current project is, for example, xx-xx-1. |
| zone | Yes | String | Country/Region |

Table 3-15 ProductInfoBean

| Parameter | Mandatory | Type | Description |
|--------------------|-----------|------------------|--------------------------------------------------------------------------------------------------|
| all_resource_names | No | Array of strings | Resource list |
| cloud_service_type | Yes | String | Cloud service type |
| display_id | No | String | Display ID |
| product_id | Yes | String | Product ID |
| product_spec_desc | No | String | Product specification description |
| resource_name | No | String | Resource name |
| resource_size | Yes | Integer | Number of databases supported by the product or the number of OBS scans supported by the product |

| Parameter | Mandatory | Type | Description |
|--------------------------|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| resource_size_measure_id | Yes | Integer | Resource capacity measurement ID. The example values are as follows: 15: mbps (used when bandwidth is purchased), 17: gb (used when EVS disks are purchased), 14: number/time. |
| resource_spec_code | Yes | String | Product code |
| resource_type | Yes | String | Resource type |
| usage_factor | No | String | Coefficient in use |
| usage_measure_id | No | Integer | Capacity measurement ID in use |
| usage_value | No | Double | Value in use |

Response Parameters

Status code: 200

Table 3-16 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------|
| order_id | String | Order ID |

Status code: 400

Table 3-17 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Purchase the hws.service.type.sdg cloud service instance for two months for the xxxxxxxxxxxx project in China. This product supports the scanning of XX database or OBS instances.

```
POST /v1/{project_id}/period/order
{
```

```

"charging_mode" : 0,
"cloud_service_type" : "hws.service.type.sdg",
"is_auto_renew" : 0,
"period_num" : 1,
"period_type" : 2,
"region_id" : "xxxxxxxxxxxx",
"zone" : "CH",
"product_infos" : [ {
  "cloud_service_type" : "hws.service.type.sdg",
  "product_id" : "xxxxxxxxxxxx",
  "resource_size" : 30,
  "resource_size_measure_id" : 30,
  "resource_spec_code" : "base_professional",
  "resource_type" : "hws.resource.type.dsc.base"
} ]
}

```

Example Responses

Status code: 200

OK

```

{
  "order_id" : "xxxxxxxxxxxx"
}

```

Status code: 400

Invalid request.

```

{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}

```

Status Codes

| Status Code | Description |
|-------------|------------------|
| 200 | OK |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.2.2 Querying Resource Provisioning Information

Function

This API is used to query resource provisioning information and query order details based on the project ID.

URI

GET /v1/{project_id}/period/product/specification

Table 3-18 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-19 Response body parameters

| Parameter | Type | Description |
|------------|---------------------------------------------------|-------------|
| orderInfos | Array of ProductOrderInfo objects | Order list |

Table 3-20 ProductOrderInfo

| Parameter | Type | Description |
|-------------|------------------------------------|--------------------------|
| tenantId | String | Tenant ID |
| periodType | String | Subscription period type |
| periodNum | Integer | Subscription periods |
| resourceId | String | Resource ID |
| productInfo | ProductInfo object | Product details. |

Table 3-21 ProductInfo

| Parameter | Type | Description |
|------------------|------------------|---------------------|
| allResourceNames | Array of strings | Resource list. |
| cloudServiceType | String | Cloud service type. |
| displayId | String | Display ID. |
| productId | String | Product ID. |

| Parameter | Type | Description |
|---------------------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| productSpecDesc | String | Product specification description. |
| resourceName | String | Resource name. |
| resourceSize | Integer | Number of databases supported by the product or the number of OBS scans supported by the product. |
| resourceSizeMeasurementId | Integer | Resource capacity measurement ID. The example values are as follows: 15: mbps (used when bandwidth is purchased), 17: gb (used when EVS disks are purchased), 14: number/time. |
| resourceSpecCode | String | Product code. |
| resourceType | String | Resource type. |
| usageFactor | String | Coefficient in use. |
| usageMeasurementId | Integer | Capacity measurement ID in use. |
| usageValue | Double | Value in use. |

Status code: 400

Table 3-22 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query resource provisioning information.

```
GET /v1/{project_id}/period/product/specification
```

Example Responses

Status code: 200

Request sent

```
{
  "orderInfos" : [ {
```

```

"tenantId" : "xxxxxxxxxxxx",
"periodType" : 2,
"periodNum" : 1,
"resourceId" : "xxxxxxxxxxxx",
"productInfo" : {
  "cloudServiceType" : "hws.service.type.sdg",
  "productId" : "xxxxxxxxxxxx",
  "resourceSize" : 30,
  "resourceSizeMeasureId" : 30,
  "resourceSpecCode" : "base_professional",
  "resourceType" : "hws.resource.type.dsc.base"
}
}
}

```

Status code: 400

Invalid request.

```

{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}

```

Status Codes

| Status Code | Description |
|-------------|------------------|
| 200 | Request sent |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.3 Dynamic Data Masking

3.3.1 Masking Sensitive Data

Function

This API is used to mask sensitive data.

URI

POST /v1/{project_id}/data/mask

Table 3-23 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-24 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-25 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------------|-----------|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| mask_strategies | Yes | Array of MaskStrategies objects | List of sensitive data masking policies. The number of masking policies in a list cannot exceed 100. Each policy corresponds to a field. |
| data | Yes | Array of Map<String, Object> objects | Data list |

Table 3-26 MaskStrategies

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|------------------------------------------------------------|
| name | Yes | String | Field name, which can contain a maximum of 256 characters. |

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| algorithm | Yes | String | Name of a sensitive data masking algorithm. For details, see "Configuring a Dynamic Sensitive Data Masking Policy" in the appendix. Enumeration values: <ul style="list-style-type: none"> • SHA256 • SHA512 • AES • PRESNM • MASKNM • PRESXY • MASKXY • SYMBOL • KEYWORD • NUMERIC |
| parameters | No | Map<String, Object> | Name of a sensitive data masking algorithm. For details, see "Dynamically Configuring a Sensitive Data Masking Policy" in the appendix. |

Response Parameters

Status code: 200

Table 3-27 Response body parameters

| Parameter | Type | Description |
|-------------|--------------------------------------|-----------------------------------------------------------------------------|
| masked_data | Array of Map<String, Object> objects | List of masked data. The data structure is the same as that in the request. |

Status code: 400

Table 3-28 Response body parameters

| Parameter | Type | Description |
|------------|--------|-------------|
| error_code | String | Error Code |

| Parameter | Type | Description |
|-----------|--------|---------------|
| error_msg | String | Error Message |

Example Requests

Specify in the masking policy list to use the KEYWORD masking algorithm to mask the "col" field by replacing the specified keyword with "target".

POST https://{endpoint}/v1/{project_id}/data/mask

```
{
  "mask_strategies" : {
    "name" : "col",
    "algorithm" : "KEYWORD",
    "parameters" : {
      "key" : "keyword",
      "target" : "target"
    }
  },
  "data" : {
    "col" : "keyword"
  }
}
```

Example Responses

Status code: 200

Sensitive data masked.

```
{
  "masked_data" : [ {
    "col" : "target"
  } ]
}
```

Status code: 400

Invalid request.

```
{
  "error_code" : "DSC.00000004",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|------------------------|
| 200 | Sensitive data masked. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.4 Alarm Notifications

3.4.1 Querying an Alarm Notification Topic

Function

This API is used to query alarm notification topics. The default topic, number of confirmed topics, and list are returned.

URI

GET /v1/{project_id}/sdg/smn/topics

Table 3-29 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-30 Query Parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page limit |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-31 Response body parameters

| Parameter | Type | Description |
|-------------------|---------|-----------------------------------------------------|
| id | String | SMN topic ID |
| default_topic_urn | String | Unique resource identifier of the default SMN topic |
| topic_count | Integer | Number of confirmed SMN topics |

| Parameter | Type | Description |
|-----------|--------------------------------------------|------------------------------|
| topics | Array of TopicBean objects | List of confirmed SMN topics |

Table 3-32 TopicBean

| Parameter | Type | Description |
|-----------|--------|--------------------------------------------|
| name | String | SMN topic name |
| topic_urn | String | Unique resource identifier of an SMN topic |

Status code: 400

Table 3-33 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query an alarm notification topic.

```
GET /v1/{project_id}/sdg/smn/topics
```

Example Responses

Status code: 200

Request sent

```
{
  "id": "xxxxxx",
  "default_topic_urn": "xxxxxx",
  "topic_count": 1,
  "topics": [ {
    "name": "xxxxxx",
    "topic_urn": "xxxxxx"
  } ]
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.4.2 Modifying an Alarm Notification Topic

Function

This API is used to modify general alarm notification settings, such as the associated project ID, notification topic, and notification status (0: notification disabled; 1: notification enabled).

URI

PUT /v1/{project_id}/sdg/smn/topic

Table 3-34 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-35 Request body parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------|---------------------------------------------------------------|
| id | Yes | String | DSC alarm topic ID (Not Simple Message Notification topic ID) |
| project_id | No | String | Project ID |
| status | No | Integer | Alarm notification status |
| topic_urn | Yes | String | Unique resource identifier of an SMN topic |

Response Parameters

Status code: 200

Table 3-36 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-37 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Change the notification topic to xxxxxxxxxxxxxxxxxxxx, Project ID to xxxxxxxxxxxxxxxxxxxx, and set the alarm notification status to Enabled.

```
PUT /v1/{project_id}/sdg/smn/topic
{
  "id": "xxxxxxxxxxxxxxxxxxxx",
  "project_id": "xxxxxxxxxxxxxxxx",
  "status": 1,
  "topic_urn": "xxxxxxxxxxxxxxxx"
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg": "xxx",
  "status": "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.5 Image Watermarking

3.5.1 Injecting Invisible Watermarks into Images

Function

This API is used to inject text watermarks or image watermarks into an image. You need to pass an image in formData format to this API. DSC returns the binary stream of the watermarked image. Currently, the supported image formats include *.jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

POST /v1/{project_id}/image/watermark/embed

Table 3-38 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-39 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-40 FormData parameters

| Parameter | Mandatory | Type | Description |
|-----------------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| file | Yes | File | Image into which watermarks are to be injected. The width and height must be greater than 512 pixels. |
| blind_watermark | No | String | Content of the invisible text watermark, which cannot exceed 32 characters. Currently, only digits and uppercase and lowercase letters are supported. It is used as an alternative of image_watermark. |
| image_watermark | No | File | Invisible image watermark file to be injected. It is used as an alternative of blind_watermark. |

Response Parameters

Status code: 400

Table 3-41 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Add the invisible text watermark "testWaterMark" to image "test.PNG".

NOTE

The request is submitted through a form. In the request, "file" indicates a specific file.

```
POST /v1/{project_id}/image/watermark/embed HTTP/1.1
```

```
{
  "blind_watermark" : "testWaterMark",
  "file" : "test.PNG"
}
```

Example Responses

Status code: 200

Request successful.

```
"{"Watermarked image\}"
```

Status code: 400

Invalid request.

```
{  
  "error_code" : "DSC.00000007",  
  "error_msg" : "File format error"  
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.5.2 Extracting Invisible Text Watermarks

Function

This API is used to extract text watermarks from images. You need to pass an image in formData format to this API. DSC returns the extracted invisible text watermarks in JSON format. Currently, the supported image formats include .jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

POST /v1/{project_id}/image/watermark/extract

Table 3-42 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-43 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-44 FormData parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------|
| file | Yes | File | Image from which invisible watermarks are to be extracted. |
| mark_len | No | String | Length of the watermark to be extracted. The length of mark_len is greater than 0 and less than 32. |

Response Parameters

Status code: 200

Table 3-45 Response body parameters

| Parameter | Type | Description |
|-----------|--------|--------------------------------------------------------------|
| watermark | String | Invisible watermark. The length cannot exceed 32 characters. |

Status code: 400

Table 3-46 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible text watermark from image "test.PNG".

NOTE

The request is submitted through a form. In the request, "file" indicates a specific file.

```
POST /v1/{project_id}/image/watermark/extract HTTP/1.1
{
  "file": "test.PNG"
}
```

Example Responses

Status code: 200

Request successful.

```
{
  "watermark": "mark!"
}
```

Status code: 400

Invalid request.

```
{
  "error_code": "DSC.00000007 ",
  "error_msg": "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.5.3 Extracting Invisible Watermarks from Images

Function

This API is used to extract invisible image watermarks from images. You need to pass a watermarked image in formData format to this API. DSC returns the binary stream of the extracted invisible watermark. Currently, the supported image formats

include .jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

POST /v1/{project_id}/image/watermark/extract-image

Table 3-47 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-48 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token. It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-49 FormData parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|------|-----------------------------------------------------------------|
| file | Yes | File | Image file from which invisible watermarks are to be extracted. |

Response Parameters

Status code: 400

Table 3-50 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible watermark from image "test.PNG".

NOTE

The request is submitted through a form. In the request, "file" indicates a specific file.

```
POST /v1/{project_id}/image/watermark/extract-image HTTP/1.1
```

```
{  
  "file" : "test.PNG"  
}
```

Example Responses

Status code: 200

Request succeeded.

```
"{\image\}"
```

Status code: 400

Invalid request.

```
{  
  "error_code" : "DSC.00000007 ",  
  "error_msg" : "File format error"  
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.5.4 Injecting Invisible Watermarks into Images (Image Addresses)

Function

This API is used to inject invisible text watermarks or invisible image watermarks into an image whose storage address is specified (only a Huawei Cloudcloud service OBS path is supported). DSC returns the injected watermark in a specified storage location (an OBS path). The supported image formats include .jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

```
POST /v1/{project_id}/image-address/watermark/embed
```


Table 3-51 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-52 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token. It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-53 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | Yes | String | ID of the region where the current project is located, for example, xx-xx-1. |
| src_file | Yes | String | Address of the image to be watermarked. Currently, only Huawei Cloudcloud service OBS files are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png. |
| blind_watermark | No | String | Content of the invisible text watermark, which cannot exceed 32 characters. Currently, only digits and uppercase and lowercase letters are supported. Either this parameter or image_watermark must be set. |

| Parameter | Mandatory | Type | Description |
|-----------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| image_watermark | No | String | Address of the invisible image watermark. Its format requirements are the same as those of src_file. Either image_watermark or blind_watermark must be set. If both are set, only image_watermark takes effect. |
| dst_file | No | String | Storage address of the watermarked image. The format and requirements are the same as those of the src_file field. If this field is not set, the value of src_file is used by default, that is, the original file is overwritten after the watermark is added. |

Response Parameters

Status code: 200

Table 3-54 Response body parameters

| Parameter | Type | Description |
|------------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | String | ID of the region where the current project is located, for example, xx-xx-1. |
| watermarked_file | String | Address of the watermarked image on OBS. Currently, only Huawei Cloud service OBS files are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png. |

Status code: 400

Table 3-55 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Embed the invisible watermark "testWaterMark" into the image at obs://hwbucket/test.png in the xx-xx-1 region.

```
POST /v1/{project_id}/image-address/watermark/embed HTTP/1.1
{
  "region_id": "xx-xx-1",
  "src_file": "obs://hwbucket/test.png",
  "blind_watermark": "testWaterMark"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "region_id": "xx-xx-1",
  "watermarked_file": "obs://hwbucket/test.png"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "DSC.00000007",
  "error_msg": "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.5.5 Extracting Dark Watermarks from Images (Image Addresses)

Function

This API is used to extract invisible text watermarks from an image whose storage address is specified (only a Huawei Cloudcloud service OBS path is supported). The supported image formats

include .jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

POST /v1/{project_id}/image-address/watermark/extract

Table 3-56 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-57 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-58 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|----------------------------------------------------------------------|
| region_id | Yes | String | ID of the region where the project is located, for example, xx-xx-1. |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| src_file | Yes | String | Address of the image from which the text watermark is to be extracted. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png, where obs:// indicates OBS, hwbucket indicates the bucket name, and hwinfo/hw.png indicates the full path name of the object. |
| mark_len | No | Integer | Length of the watermark to be extracted. The value ranges from 0 to 32. This parameter improves watermark extraction performance. |

Response Parameters

Status code: 200

Table 3-59 Response body parameters

| Parameter | Type | Description |
|-----------|--------|--------------------------------------------------------------|
| watermark | String | Invisible watermark. The length cannot exceed 32 characters. |

Status code: 400

Table 3-60 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible text watermarks from the image at obs://hwbucket/hwinfo/hw.png.

```
POST /v1/{project_id}/image-address/watermark/extract HTTP/1.1
{
  "region_id" : "xx-xx-1",
  "src_file" : "obs://hwbucket/hwinfo/hw.png"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "watermark" : "mark!"
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "DSC.00000007 ",
  "error_msg" : "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.5.6 Extracting Invisible Image Watermarks from Images (Image Addresses)

Function

This API is used to extract invisible image watermarks from images whose storage address is specified (only a Huawei Cloudcloud service OBS path is supported).DSC returns the watermark images in a specified storage location (an OBS path). The supported image format is .jpg, .jpeg, .jpe, .png, .bmp, .dib, .rle, .tiff, .tif, .ppm, .webp, .tga, .tpic, and .gif.

URI

POST /v1/{project_id}/image-address/watermark/extract-image

Table 3-61 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-62 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-63 Request body parameters

| Parameter | Mandatory | Type | Description |
|------------------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | Yes | String | ID of the region where the project is located, for example, xx-xx-1. |
| src_file | Yes | String | Address of the image from which the image watermark is to be extracted. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png. |
| image_water mark | Yes | String | Storage address of the extracted image watermark. The format requirements are the same as those of src_file. |

Response Parameters

Status code: 200

Table 3-64 Response body parameters

| Parameter | Type | Description |
|-----------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | String | ID of the region where the current project is located, for example, xx-xx-1. |
| image_watermark | String | Storage address of the extracted image watermark. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png. |

Status code: 400

Table 3-65 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible watermark from the image at obs://hwbucket/hwinfo/hw.png and store the extracted watermark image at obs://hwbucket/watermarkfile/mark.png.

```
POST /v1/{project_id}/obs-image/image-watermark/extract HTTP/1.1
{
  "region_id": "xx-xx-1",
  "src_file": "obs://hwbucket/hwinfo/hw.png",
  "image_watermark": "obs://hwbucket/watermarkfile/mark.png"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "region_id": "xx-xx-1",
  "image_watermark": "obs://hwbucket/watermarkfile/mark.png"
}
```


Status code: 400

Invalid request

```
{
  "error_code" : "DSC.00000007 ",
  "error_msg" : "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.6 Asset Management

3.6.1 Editing a Data Asset Name

Function

This API is used to edit a data asset name.

URI

PUT /v1/{project_id}/sdg/asset/{asset_id}/name

Table 3-66 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| asset_id | Yes | String | Asset ID |
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-67 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-------------|
| name | No | String | Asset name |

Response Parameters

Status code: 200

Table 3-68 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-69 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Set the data asset name to xxxxxxx.

```
PUT /v1/{project_id}/sdg/asset/{asset_id}/name
{
  "name" : "xxxxxxx"
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg" : "xxxx",
  "status" : "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.6.2 Querying the Data Asset Scanning Authorization

Function

This API is used to query the data asset scanning authorization.

URI

GET /v1/{project_id}/sdg/asset/obs/buckets

Table 3-70 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-71 Query Parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------|
| added | No | Boolean | Authorized |
| offset | No | Integer | Page number |
| limit | No | Integer | Page limit |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-72 Response body parameters

| Parameter | Type | Description |
|-----------|-----------------------------------------|-----------------------------|
| buckets | Array of Bucket objects | OBS bucket list |
| total | Integer | Total number of OBS buckets |

Table 3-73 Bucket

| Parameter | Type | Description |
|-----------------|---------|-----------------|
| asset_name | String | Asset name |
| bucket_location | String | Bucket location |
| bucket_name | String | Bucket name |
| bucket_policy | String | Bucket policy |
| create_time | Long | Time created |
| deleted | Boolean | Deleted or not |
| id | String | Bucket ID |
| is_deleted | Boolean | Deleted or not |

Status code: 400**Table 3-74** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query the data asset scanning authorization.

```
GET /v1/{project_id}/sdg/asset/obs/buckets
```

Example Responses

Status code: 200

Request sent

```
{
  "buckets" : [ {
    "bucket_name" : "xxxx",
    "bucket_location" : "xxxx",
    "create_time" : 1667379757698,
    "is_deleted" : false
  } ],
  "total" : 1
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.6.3 Adding a Data Asset Scanning Authorization

Function

This API is used to add a data asset scanning authorization.

URI

POST /v1/{project_id}/sdg/asset/obs/buckets

Table 3-75 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-76 Query Parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-------------|
| type | No | String | Asset type |

Request Parameters

Table 3-77 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|------------------------------------|-----------------|
| buckets | No | Array of BucketBean objects | OBS bucket list |

Table 3-78 BucketBean

| Parameter | Mandatory | Type | Description |
|---------------|-----------|--------|-----------------|
| asset_name | No | String | Asset name |
| location | No | String | Bucket location |
| bucket_name | No | String | Bucket name |
| bucket_policy | No | String | Bucket policy |

Response Parameters

Status code: 200

Table 3-79 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-80 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Add data asset scan authorization for bucket xxxx.

```
POST /v1/{project_id}/sdg/asset/obs/buckets
{
  "buckets": [ {
    "asset_name": "xxxx",
    "location": "xxxx",
    "bucket_name": "xxxx",
    "bucket_policy": "private"
  } ]
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg": "xxxx",
  "status": "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.6.4 Deleting a Data Asset Scanning Authorization

Function

This API is used to delete a data asset scanning authorization.

URI

DELETE /v1/{project_id}/sdg/asset/obs/bucket/{bucket_id}

Table 3-81 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| bucket_id | Yes | String | Bucket ID |
| project_id | Yes | String | Project ID |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-82 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-83 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Delete a data asset scanning authorization.

```
DELETE /v1/{project_id}/sdg/asset/obs/bucket/{bucket_id}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg" : "xxx",
  "status" : "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{  
  "error_code" : "dsc.40000011",  
  "error_msg" : "Invalid parameter"  
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.7 Document Watermarking

3.7.1 Injecting Watermarks into Documents

Function

This API is used to inject invisible text watermarks, visible text watermarks, or visible image watermarks into Word (.docx), PPT (.pptx), Excel (.xlsx), and PDF (.pdf) files. You need to pass file and watermark information in formData format to this API, DSC returns the binary stream of the watermarked file.

URI

POST /v1/{project_id}/sdg/doc/watermark/embed

Table 3-84 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-85 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-86 FormData parameters

| Parameter | Mandatory | Type | Description |
|----------------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| doc_type | Yes | String | Type of the document into which a watermark is injected. Enumeration values: <ul style="list-style-type: none"> • WORD • EXCEL • PDF • PPT |
| file_password | No | String | This API is used to read file passwords, which can contain a maximum of 256 characters. If an Office Word document requires a password for read or domain control, you need to enter the password to open the file. |
| marked_file_password | No | String | This API is used to set a password for the file after the watermark is added. The maximum length is 256 characters. The file is not encrypted by default and does not require a password. |
| readonly_password | No | String | This API is used to set a password for reading the file after the watermark is added. The maximum length is 256 characters. By default, the file is not encrypted for reading and does not require a password. |
| visible_watermark | No | String | Visible watermark content. |

| Parameter | Mandatory | Type | Description |
|-----------------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| font_size | No | String | Font size of a visible watermark. The value ranges from 1 to 100, and the default value is 50. |
| rotation | No | String | Font angle of a visible watermark, in anticlockwise direction. The value range is [0, 90], and the default value is 45. |
| opacity | No | String | Transparency of a visible watermark. The value range is [0, 1], and the default value is 0.3. |
| blind_watermark | No | String | Invisible watermark content. |
| file | Yes | File | Document into which a watermark is injected. |
| image_mark | No | File | Image added in the document as the watermark. The image must be in the PNG or JPG format and cannot exceed 1 MB. The value of name in Content-Disposition in the API request must be set to image_mark. |
| visible_type | No | String | Type of visible watermark. The default value of this field is TEXT. When this field is set to IMAGE: <ul style="list-style-type: none">the image must be in the PNG or JPG format and cannot exceed 1 MB,the value of name in Content-Disposition in the API request must be set to image_mark,the fields visible_watermark, font_size, rotation, and opacity will not take effect. Enumeration values: <ul style="list-style-type: none">TEXTIMAGE |

Response Parameters

Status code: 400

Table 3-87 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

NOTE

Submit the request using the form. The "file" indicates a specified document.

```
POST /v1/{project_id}/sdg/doc/watermark/embed
{
  "file": "test.doc",
  "doc_type": "WORD",
  "opacity": "0.1",
  "font_size": "30",
  "rotation": "45",
  "blind_watermark": "blind_watermark",
  "visible_watermark": "visible_watermark"
}
```

Example Responses

Status code: 200

Request successful.

```
"{"Watermarked document"}"
```

Status code: 400

Invalid request.

```
{
  "error_code": "DSC.00000007 ",
  "error_msg": "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.7.2 Extracting Invisible Watermarks

Function

This API is used to extract invisible watermarks from Word (.docx), PPT (.pptx), Excel (.xlsx), and PDF (.pdf) files. You need to pass watermarked formData files to this API, DSC returns extracted text watermarks in JSON format.

URI

POST /v1/{project_id}/sdg/doc/watermark/extract

Table 3-88 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-89 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token, which can be obtained by calling the IAM API (value of X-Subject-Token in the response header). |

Table 3-90 FormData parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| doc_type | Yes | String | Type of the document from which a watermark needs to be extracted. Enumeration values: <ul style="list-style-type: none">• WORD• EXCEL• PDF• PPT |

| Parameter | Mandatory | Type | Description |
|---------------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| file_password | No | String | Password for opening a file, which can contain a maximum of 256 characters. Opening a watermarked file does not require a password. If an Office Word document requires a password for read or domain control, you need to enter the password to open the file. |
| file | Yes | File | File to be uploaded for watermark extraction. |

Response Parameters

Status code: 200

Table 3-91 Response body parameters

| Parameter | Type | Description |
|-----------|--------|--------------------------------------------------------------|
| watermark | String | Invisible watermark. The length cannot exceed 32 characters. |

Status code: 400

Table 3-92 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible watermark of the testMarked.doc file whose document type is word.

NOTE

The request is submitted through a form. In the request, "file" indicates a specific file.

```
POST /v1/{project_id}/sdg/doc/watermark/extract
```

```
{
  "file" : "testMarked.doc",
```

```
"doc_type" : "WORD"  
}
```

Example Responses

Status code: 200

Request successful.

```
{  
  "watermark" : "mark!"  
}
```

Status code: 400

Invalid request.

```
{  
  "error_code" : "DSC.00000007",  
  "error_msg" : "File format error"  
}
```

Status Codes

| Status Code | Description |
|-------------|---------------------|
| 200 | Request successful. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.7.3 Injecting Watermarks into Documents (Document Addresses)

Function

This API is used to inject visible text watermarks, invisible text watermarks, or visible image watermarks into Word (.docx), PPT (.pptx), Excel (.xlsx), and PDF (.pdf)* documents. You need to pass a document address (currently, only OBS is supported) and watermark information to this API. DSC returns the storage address of the watermarked document.

URI

POST /v1/{project_id}/doc-address/watermark/embed

Table 3-93 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-94 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token. It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-95 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | Yes | String | ID of the region where the project is located, for example, xx-xx-1. |
| src_file | Yes | String | Address of the document to which the watermark is to be added. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object , where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.png. |
| doc_type | Yes | String | Type of the document to be watermarked Enumeration values: <ul style="list-style-type: none"> • WORD • EXCEL • PDF • PPT |

| Parameter | Mandatory | Type | Description |
|-------------------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dst_file | No | String | Storage address of the watermarked document. The format and requirements are the same as those of the src_file field. If this field is not set, the value of src_file is used by default. In this case, the original file is overwritten after the watermark is added. |
| blind_watermark | No | String | Content of the invisible text watermark. Either blind_watermark or visible_watermark must be set. |
| visible_watermark | No | String | Content of the visible text watermark. Either blind_watermark or visible_watermark must be set. |
| image_mark | No | String | Address of document to be injected with a visible image watermark. The field format must be the same as that of src_file. The image file must be in PNG or JPG format. Otherwise, an error is returned. The image file size cannot exceed 1 MB. |
| visible_type | No | String | Whether the visible watermark is of the text or image type. If the default value TEXT is used, you need to set visible_watermark to specify a visible text watermark. If this parameter is set to IMAGE, you need to set image_watermark to specify the address of an image watermark. In this case, the visible_watermark, font_size, rotation, and opacity fields will not take effect. Enumeration values: <ul style="list-style-type: none">• TEXT• IMAGE |

| Parameter | Mandatory | Type | Description |
|----------------------|-----------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| file_password | No | String | Password for reading the file to be watermarked, which can contain a maximum of 256 characters. If an Office Word document requires a password for read or domain control, you need to enter the password to open the file. |
| marked_file_password | No | String | Password for a watermarked document, which can contain a maximum of 256 characters. By default, the document does not have a password. |
| readonly_password | No | String | Read-only password for a watermarked document, which can contain a maximum of 256 characters. By default, the document does not have a read-only password. |
| front | No | Integer | Font size of a visible watermark. The value range is [1, 100], and the default value is 50. |
| rotation | No | Integer | Font angle of a visible watermark, in anticlockwise direction. The value range is [0, 90], and the default value is 45. |
| opacity | No | Float | Transparency of a visible watermark. The value range is [0, 1], and the default value is 0.3. |

Response Parameters

Status code: 200

Table 3-96 Response body parameters

| Parameter | Type | Description |
|-----------|--------|------------------------------------------------------------------------------|
| region_id | String | ID of the region where the current project is located, for example, xx-xx-1. |

| Parameter | Type | Description |
|------------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| watermarked_file | String | Address of the document to which the watermark is to be added. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object , where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.doc. |

Status code: 400

Table 3-97 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Embed the invisible watermark "blind_watermark" and visible watermark "visible_watermark" to the Word document at obs://hwbucket/hwinfo/hw.doc.

```
POST /v1/{project_id}/doc-address/watermark/embed
{
  "region_id" : "xx-xx-1",
  "src_file" : "obs://hwbucket/hwinfo/hw.doc",
  "doc_type" : "WORD",
  "blind_watermark" : "blind_watermark",
  "visible_watermark" : "visible_watermark"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "region_id" : "xx-xx-1",
  "watermarked_file" : "obs://hwbucket/hwinfo/hw.docx"
}
```

Status code: 400

Invalid request.

```
{
  "error_code" : "DSC.00000007 ",
  "error_msg" : "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.7.4 Extracting Invisible Watermarks from Documents (Document Addresses)

Function

This API is used to extract invisible text watermarks from Word (.docx), PPT (.pptx), Excel (.xlsx), and PDF (.pdf) documents. You need to pass the address of a watermarked document (only OBS path is supported) to this API. DSC returns the invisible text watermark extracted from the document in JSON format.

URI

POST /v1/{project_id}/doc-address/watermark/extract

Table 3-98 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID. |

Request Parameters

Table 3-99 Request header parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| X-Auth-Token | Yes | String | User token. It can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is a token. |

Table 3-100 Request body parameters

| Parameter | Mandatory | Type | Description |
|---------------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| region_id | Yes | String | ID of the region where the project is located, for example, xx-xx-1. |
| doc_type | Yes | String | Type of the document whose watermark is to be extracted. Enumeration values: <ul style="list-style-type: none">• WORD• EXCEL• PDF• PPT |
| src_file | Yes | String | Address of the document whose text watermark is to be extracted. Currently, only Huawei Cloudcloud service OBS objects are supported. The format is obs://bucket/object, where bucket indicates the name of the OBS bucket in the same region as the current project, and object indicates the full path name of the object. For example, obs://hwbucket/hwinfo/hw.doc. |
| file_password | No | String | Password for opening a file, which can contain a maximum of 256 characters. If an Office Word document requires a password for read or domain control, you need to enter the password to open the file. |

Response Parameters

Status code: 200

Table 3-101 Response body parameters

| Parameter | Type | Description |
|-----------|--------|--------------------------------------------------------------|
| watermark | String | Invisible watermark. The length cannot exceed 32 characters. |

Status code: 400

Table 3-102 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Extract the invisible watermark from the Word document at obs://hwbucket/hwinfo/hw.docx.

```
POST /v1/{project_id}/doc-address/watermark/extract
{
  "region_id": "xx-xx-1",
  "src_file": "obs://hwbucket/hwinfo/hw.docx",
  "doc_type": "WORD"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "watermark": "blind_watermark"
}
```

Status code: 400

Invalid request.

```
{
  "error_code": "DSC.00000007",
  "error_msg": "File format error"
}
```

Status Codes

| Status Code | Description |
|-------------|--------------------|
| 200 | Request succeeded. |
| 400 | Invalid request. |

Error Codes

See [Error Codes](#).

3.8 Sensitive Data Discovery

3.8.1 Querying the Identification Task List

Function

This API is used to query the identification task list.

URI

GET /v1/{project_id}/sdg/scan/jobs

Table 3-103 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-104 Query Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------|------------------------------------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page size |
| content | No | String | Content |
| start_time | No | String | This parameter is currently unavailable. |
| end_time | No | String | This parameter is currently unavailable. |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-105 Response body parameters

| Parameter | Type | Description |
|-----------|------------------------------------------|------------------------------|
| tasks | Array of ScanJob objects | List of identification tasks |
| total | Long | Total number of tasks |

Table 3-106 ScanJob

| Parameter | Type | Description |
|----------------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id | String | Task ID |
| name | String | Task name |
| rule_groups | Array of strings | Rule group used by a task |
| cycle | String | Task execution type Enumeration values: <ul style="list-style-type: none">• ONCE• DAY• WEEK• MONTH |
| status | String | Task status Enumeration values: <ul style="list-style-type: none">• INIT• WAITING• RUNNING• FAILED• STOPPED• FINISHED• TERMINATED |
| last_run_time | Long | Last execution time |
| create_time | Long | Task creation time |
| last_scan_risk | String | Risk level according to the last identification |
| use_nlp | Boolean | Whether NLP is used for identification |
| open | Boolean | Whether the task is started |
| topic_urn | String | SMN topic |
| start_time | Long | Task start time |

Status code: 400**Table 3-107** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Querying the Identification Task List

```
GET /v1/{project_id}/sdg/scan/jobs
```

Example Responses

Status code: 200

OK

```
{
  "total": 1,
  "tasks": [ {
    "id": "xxxxxxx",
    "name": "ScanDemo",
    "cycle": "ONCE",
    "status": "FINISHED",
    "open": true,
    "rule_groups": [ "PCI" ],
    "last_run_time": 1634612489173,
    "create_time": 1630982438506,
    "last_scan_risk": "HIGH",
    "use_nlp": false,
    "topic_urn": "",
    "start_time": 1630983532673
  } ]
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | OK |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.2 Querying the Result of an Identification Task

Function

This API is used to query the result of a specified identification task.

URI

GET /v1/{project_id}/sdg/scan/job/{job_id}/results

Table 3-108 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |
| job_id | Yes | String | Task ID |

Table 3-109 Query Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page size |
| type | No | String | Asset type Enumeration values: <ul style="list-style-type: none"> • DATABASE • OBS • BIGDATA |
| start_time | No | String | Starting time (This parameter is currently unavailable) |
| end_time | No | String | End time (This parameter is currently unavailable) |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-110 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------|
| job_id | String | Task ID |
| job_name | String | Task name |
| type | String | Asset type |

| Parameter | Type | Description |
|-----------------|--------------------------------------|-----------------------------------------------------|
| db_scan_result | DbScanResult object | Result of identification tasks for databases |
| obs_scan_result | ObsScanResult object | Result of the identification task for OBS |
| es_scan_result | EsScanResult object | Result of the identification task for Elasticsearch |

Table 3-111 DbScanResult

| Parameter | Type | Description |
|-----------------|---------------------------------------------------|---------------------------------------------------|
| total | Integer | Total number of results |
| db_scan_results | Array of DbScanResultInfo objects | Result list of identification tasks for databases |

Table 3-112 DbScanResultInfo

| Parameter | Type | Description |
|---------------------|----------------------------------------------|--------------------------------|
| task_id | String | Task ID |
| db_name | String | Database name |
| table_id | String | Table ID |
| table_name | String | Table name |
| risk_level | Integer | Risk level |
| sensitive_data_type | Array of strings | Matched rules |
| match_info | Array of DbMatchInfo objects | Matched rules of table columns |

Table 3-113 DbMatchInfo

| Parameter | Type | Description |
|-------------|--------|--------------------------|
| column_name | String | Column name |
| rule_name | String | Name of the matched rule |
| rule_id | String | ID of the matched rule |

| Parameter | Type | Description |
|-----------------|----------------|--------------------------------|
| rule_risk_level | Integer | Risk level of the matched rule |
| column_line | Array of longs | Columns of risk data |

Table 3-114 ObsScanResult

| Parameter | Type | Description |
|-----------------|----------------------------------------------------|------------------------------------------------|
| total | Integer | Total number of results |
| db_scan_results | Array of ObsScanResultInfo objects | Result list of the identification task for OBS |

Table 3-115 ObsScanResultInfo

| Parameter | Type | Description |
|---------------------|------------------|----------------------|
| task_id | String | Task ID |
| bucket_id | String | OBS bucket ID |
| bucket_name | String | OBS bucket name |
| file_path | String | Directory |
| file_name | String | File name |
| md5 | String | File MD5 |
| risk_level | Integer | Risk level |
| sensitive_data_type | Array of strings | Sensitive data types |

Table 3-116 EsScanResult

| Parameter | Type | Description |
|-----------------|---------------------------------------------------|----------------------------------------------------------|
| total | Integer | Total number of results |
| db_scan_results | Array of EsScanResultInfo objects | Result list of the identification task for Elasticsearch |

Table 3-117 EsScanResultInfo

| Parameter | Type | Description |
|---------------------|----------------------------------------------|--------------------------|
| task_id | String | Task ID |
| index_name | String | Index name |
| type_id | String | Type ID |
| type_name | String | Type name |
| risk_level | Integer | Risk level |
| sensitive_data_type | Array of strings | Sensitive data type |
| match_info | Array of EsMatchInfo objects | Details of matched rules |

Table 3-118 EsMatchInfo

| Parameter | Type | Description |
|-----------------|---------|--------------------------------|
| field_name | String | Data field name |
| rule_name | String | Rule name |
| rule_id | String | Rule ID |
| rule_risk_level | Integer | Risk level of the matched rule |

Status code: 400**Table 3-119** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Querying the Result of an Identification Task

```
GET /v1/{project_id}/sdg/scan/job/{job_id}/results
```

Example Responses

Status code: 200

OK

```
{
  "job_id": "xxxxxx",
  "job_name": "xxxxxx",
  "type": "DATABASE",
  "db_scan_result": {
    "total": 1,
    "db_scan_results": [ {
      "task_id": "xxxxxx",
      "db_name": "xxxxxx",
      "table_id": "xxxxxx",
      "table_name": "student",
      "risk_level": 6,
      "sensitive_data_type": [ "xxxxxx", "xxxxxx" ],
      "match_info": [ {
        "column_name": "phone",
        "rule_name": "xxxxxx",
        "rule_id": "xxxxxx",
        "rule_risk_level": 6,
        "column_line": [ 1, 3 ]
      }, {
        "column_name": "email",
        "rule_name": "xxxxxx",
        "rule_id": "xxxxxx",
        "rule_risk_level": 1,
        "column_line": [ 1, 3 ]
      } ]
    } ]
  },
  "obs_scan_result": null,
  "es_scan_result": null
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | OK |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.3 Viewing the Rule List

Function

This API is used to query the sensitive data scanning rule list. The total number of rules and the rule list are returned.

URI

GET /v1/{project_id}/sdg/server/scan/rules

Table 3-120 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-121 Query Parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page limit |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-122 Response body parameters

| Parameter | Type | Description |
|-----------|-----------------------------------------------|-----------------------|
| rules | Array of ResponseRule objects | Rule list |
| total | Integer | Total number of rules |

Table 3-123 ResponseRule

| Parameter | Type | Description |
|----------------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| category | String | Rule type, which can be built-in rule (BUILT_IN) or self-built rule (BUILT_SELF). Enumeration values: <ul style="list-style-type: none">• BUILT_IN• BUILT_SELF |
| delete_allowed | Boolean | Whether deletion is allowed |

| Parameter | Type | Description |
|----------------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| group_names | String | Rule group |
| id | String | Rule ID |
| logic_operator | String | Logical operators: "AND", "OR", and "REGEX". |
| min_match | Integer | Minimum matching times |
| risk_level | Integer | Risk level |
| rule_content | String | Rule content |
| rule_desc | String | Rule description |
| rule_name | String | Rule name |
| rule_type | String | Rule type, which can be keyword (KEYWORD), regular expression (REGEX), or natural language (NLP). Enumeration values: <ul style="list-style-type: none"> • KEYWORD • REGEX • NLP |
| selected | Boolean | Whether the parameter is optional |

Status code: 400

Table 3-124 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query the sensitive data scanning rule list.

```
GET /v1/{project_id}/sdg/server/scan/rules
```

Example Responses

Status code: 200

Request sent

```
{
  "total": 1,
  "rules": [ {
```



```

"category" : "BUILT_SELF",
"delete_allowed" : true,
"group_names" : "xxxx",
"id" : "xxxxxxxxxxxx",
"logic_operator" : "AND",
"min_match" : 1,
"risk_level" : 1,
"rule_content" : "xxxx",
"rule_desc" : "xxxx",
"rule_name" : "xxxx",
"rule_type" : "KEYWORD",
"selected" : true
}]
}

```

Status code: 400

Invalid request

```

{
"error_code" : "dsc.40000011",
"error_msg" : "Invalid parameter"
}

```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.4 Creating a Sensitive Data Scanning Rule

Function

This API is used to create a sensitive data scanning rule using the parameters such as rule name, rule type, risk level, and minimum number of matching times.

URI

POST /v1/{project_id}/sdg/server/scan/rules

Table 3-125 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-126 Request body parameters

| Parameter | Mandatory | Type | Description |
|----------------|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| category | Yes | String | Rule type, which can be built-in rule (BUILT_IN) or self-built rule (BUILT_SELF). Enumeration values: <ul style="list-style-type: none"> • BUILT_IN • BUILT_SELF |
| id | No | String | Rule ID |
| logic_operator | Yes | String | Logical operators: "AND", "OR", and "REGEX". |
| min_match | Yes | Integer | Minimum matching times |
| risk_level | Yes | Integer | Risk level |
| rule_content | Yes | String | Rule content |
| rule_desc | No | String | Rule description |
| rule_name | Yes | String | Rule name |
| rule_type | Yes | String | Rule type, which can be keyword (KEYWORD), regular expression (REGEX), or natural language (NLP). Enumeration values: <ul style="list-style-type: none"> • KEYWORD • REGEX • NLP |

Response Parameters

Status code: 200

Table 3-127 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-128 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Create a sensitive data scanning rule.

```
POST /v1/{project_id}/sdg/server/scan/rules
{
  "category": "BUILT_SELF",
  "logic_operator": "AND",
  "min_match": 1,
  "risk_level": 1,
  "rule_content": "xxxx",
  "rule_desc": "xxxx",
  "rule_name": "xxxx",
  "rule_type": "KEYWORD"
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg": "xxxx",
  "status": "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.5 Modifying a Sensitive Data Scanning Rule

Function

This API is used to modify a sensitive data scanning rule.

URI

PUT /v1/{project_id}/sdg/server/scan/rules

Table 3-129 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-130 Request body parameters

| Parameter | Mandatory | Type | Description |
|----------------|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| category | Yes | String | Rule type, which can be built-in rule (BUILT_IN) or self-built rule (BUILT_SELF). Enumeration values: <ul style="list-style-type: none">• BUILT_IN• BUILT_SELF |
| id | Yes | String | Rule ID. |
| logic_operator | Yes | String | Logical operators: "AND", "OR", and "REGEX". |
| min_match | Yes | Integer | Minimum matching times. |
| risk_level | Yes | Integer | Risk level. |
| rule_content | Yes | String | Rule content. |
| rule_desc | No | String | Rule description. |
| rule_name | Yes | String | Rule name. |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| rule_type | Yes | String | Rule type, which can be keyword (KEYWORD), regular expression (REGEX), or natural language (NLP). Enumeration values: <ul style="list-style-type: none"> • KEYWORD • REGEX • NLP |

Response Parameters

Status code: 200

Table 3-131 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-132 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Modify a user-defined rule named xxxx, of which the logical operator is set to OR, minimum match value to 1, risk level to 1, rule content to xxxx, rule description to xxxx, and rule type to xxxx.

```
PUT /v1/{project_id}/sdg/server/scan/rules
```

```
{
  "category": "BUILT_SELF",
  "id": "xxxxxxxxxxxxxxxxxxxx",
  "logic_operator": "OR",
  "min_match": 1,
  "risk_level": 1,
  "rule_content": "xxxx",
  "rule_desc": "xxxx",
  "rule_name": "xxxx",
```

```
"rule_type" : "xxxx"  
}
```

Example Responses

Status code: 200

Request sent

```
{  
  "msg" : "xxxx",  
  "status" : "RESPONSE_SUCCESS"  
}
```

Status code: 400

Invalid request

```
{  
  "error_code" : "dsc.40000011",  
  "error_msg" : "Invalid parameter"  
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.6 Deleting a Sensitive Data Scanning Rule

Function

This API is used to delete a sensitive data scanning rule.

URI

DELETE /v1/{project_id}/sdg/server/scan/rules/{rule_id}

Table 3-133 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |
| rule_id | Yes | String | Rule ID |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-134 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-135 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Delete a specified sensitive data scanning rule.

```
DELETE /v1/{project_id}/sdg/server/scan/rules/{rule_id}
```

Example Responses

Status code: 200

Request sent

```
{  
  "msg": "xxx",  
  "status": "RESPONSE_SUCCESS"  
}
```

Status code: 400

Invalid request

```
{  
  "error_code": "dsc.40000011",  
  "error_msg": "Invalid parameter"  
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.7 Querying Sensitive Data Scanning Rule Groups

Function

This API is used to query the sensitive data scanning rule groups based on a specified project ID.

URI

GET /v1/{project_id}/sdg/server/scan/groups

Table 3-136 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-137 Query Parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page limit |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-138 Response body parameters

| Parameter | Type | Description |
|-----------|------------------------------------------------|-----------------------------|
| total | Integer | Total number of rule groups |
| groups | Array of ResponseGroup objects | Rule group list |

Table 3-139 ResponseGroup

| Parameter | Type | Description |
|----------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| category | String | Rule type, which can be built-in rule (BUILT_IN) or self-built rule (BUILT_SELF). Enumeration values: <ul style="list-style-type: none">• BUILT_IN• BUILT_SELF |
| delete_allowed | Boolean | Whether deletion is allowed |
| group_desc | String | Rule group description |
| group_name | String | Rule group name |
| id | String | Rule group ID |
| rule_names | String | Rule name |
| task_names | String | Scanning task name |
| is_default | Boolean | Default or not |

Status code: 400**Table 3-140** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query sensitive data scanning rule groups.

```
GET /v1/{project_id}/sdg/server/scan/groups
```

Example Responses

Status code: 200

Request sent

```
{
  "total": 1,
  "groups": [ {
    "id": "xxxxxxxxxxxx",
    "group_name": "xxx",
    "group_desc": "xxx",
    "category": "private",
    "rule_names": "xxx",
    "is_default": false
  } ]
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.8 Creating a Sensitive Data Scanning Rule Group

Function

This API is used to create a sensitive data scanning rule group using the specified rule group name and rule list.

URI

POST /v1/{project_id}/sdg/server/scan/groups

Table 3-141 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-142 Request body parameters

| Parameter | Mandatory | Type | Description |
|----------------|-----------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| category | No | String | Rule type, which can be built-in rule (BUILT_IN) or self-built rule (BUILT_SELF). Enumeration values: <ul style="list-style-type: none">• BUILT_IN• BUILT_SELF |
| default_status | No | Boolean | Whether it is a default rule group |
| group_desc | No | String | Rule group description |
| group_name | No | String | Rule group name |
| id | No | String | Rule group ID |
| rule_ids | No | Array of strings | IDs of rules in the group |

Response Parameters

Status code: 200

Table 3-143 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-144 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Create a self-built scanning rule group named xxxx.

```
POST /v1/{project_id}/sdg/server/scan/groups
{
  "category": "BUILT_SELF",
  "group_desc": "xxxx",
  "group_name": "xxxx",
  "rule_ids": [ "xxxxxxxxxxxxxxxxxxxx", "xxxxxxxxxxxxxxxxxxxx" ]
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg": "xxxx",
  "status": "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.9 Deleting a Sensitive Data Scanning Rule Group

Function

This API is used to delete a sensitive data scanning rule group of a specified ID.

URI

```
DELETE /v1/{project_id}/sdg/server/scan/groups/{group_id}
```

Table 3-145 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|---------------|
| project_id | Yes | String | Project ID |
| group_id | Yes | String | Rule group ID |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-146 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-147 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Delete a sensitive data scanning rule group.

```
DELETE /v1/{project_id}/sdg/server/scan/groups/{group_id}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg" : "xxx",
  "status" : "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.10 Creating a Sensitive Data Scanning Task

Function

This API is used to create a sensitive data scanning task with parameters such as task name, scanning mode, scanning period, scanning rule group, and scanning time.

URI

POST /v1/{project_id}/sdg/scan/job

Table 3-148 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Request Parameters

Table 3-149 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|------------------|---------------|
| asset_ids | No | Array of strings | Asset ID list |

| Parameter | Mandatory | Type | Description |
|----------------|-----------|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cycle | No | String | Scanning period, which can be DAY, WEEK, MONTH, or ONCE. Enumeration values: <ul style="list-style-type: none"> • ONCE • DAY • WEEK • MONTH |
| name | No | String | Scanning task name |
| open | No | Boolean | Whether to enable the task |
| rule_group_ids | No | Array of strings | Rule group ID list |
| start_time | No | Long | Start time of a scanning task |
| time_zone | No | String | Time zone |
| topic_urn | No | String | Unique resource identifier of a topic |
| use_nlp | No | Boolean | Whether to use NLP |

Response Parameters

Status code: 200

Table 3-150 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-151 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Create a scan job named xxxx, select the xxxxxxxxxxx rule group for it, and set it to Scan Once and Execute Immediately.

```
POST /v1/{project_id}/sdg/scan/job
{
  "asset_ids": [ "xxxx", "xxxx" ],
  "cycle": "ONCE",
  "name": "xxxx",
  "open": true,
  "rule_group_ids": [ "xxxx", "xxxx" ],
  "start_time": 0,
  "time_zone": 8,
  "topic_urn": "xxxxxxxxxxxx",
  "use_nlp": false
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg": "xxxx",
  "status": "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.8.11 Deleting a Sensitive Data Scanning Task

Function

Deleting a Sensitive Data Scanning Task

URI

DELETE /v1/{project_id}/sdg/scan/job/{job_id}

Table 3-152 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| job_id | Yes | String | Task ID |
| project_id | Yes | String | Project ID |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-153 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-154 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Deleting a Sensitive Data Scanning Task

```
DELETE /v1/{project_id}/sdg/scan/job/{job_id}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg" : "xxxx",
```

```
"status" : "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.9 Static Data Masking

3.9.1 Querying the Data Masking Task Execution List

Function

This API is used to query the data masking task execution list.

URI

GET /v1/{project_id}/sdg/server/mask/dbs/templates/{template_id}/tasks

Table 3-155 Path Parameters

| Parameter | Mandatory | Type | Description |
|-------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |
| template_id | Yes | String | Template ID |

Table 3-156 Query Parameters

| Parameter | Mandatory | Type | Description |
|--------------|-----------|--------|--------------|
| workspace_id | No | String | Workspace ID |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|-------------|
| offset | No | Integer | Page number |
| limit | No | Integer | Page limit |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-157 Response body parameters

| Parameter | Type | Description |
|-----------|-------------------------------------------------|-------------------------------|
| tasks | Array of DBMaskTaskInfo objects | Masking task list |
| total | Integer | Total number of masking tasks |

Table 3-158 DBMaskTaskInfo

| Parameter | Type | Description |
|------------------|---------|-------------------------|
| db_type | String | Database type |
| end_time | Long | Task end time |
| execute_line | Integer | Number of executed rows |
| id | String | Task ID |
| progress | Integer | Progress |
| run_status | String | Task status |
| start_time | Long | Task start time |
| task_template_id | String | Task template ID |
| type | String | Task type |

Status code: 400

Table 3-159 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Query the list of executed data masking tasks.

```
GET /v1/{project_id}/sdg/server/mask/dbs/templates/{template_id}/tasks
```

Example Responses

Status code: 200

OK

```
{
  "tasks": [ {
    "db_type": "MySQL",
    "end_time": 1658717568622,
    "execute_line": 100000,
    "id": "xxxxxxxxxxxx",
    "progress": 100,
    "run_status": "FINISHED",
    "start_time": 1658717544469,
    "task_template_id": "xxxxxxxxxxxx",
    "type": "MANUAL"
  } ],
  "total": 1
}
```

Status code: 400

Invalid request

```
{
  "error_code": "dsc.40000011",
  "error_msg": "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | OK |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.9.2 Starting or Stopping a Data Masking Task

Function

This API is used to start or stop a data masking task.

URI

POST /v1/{project_id}/sdg/server/mask/dbs/templates/{template_id}/operation

Table 3-160 Path Parameters

| Parameter | Mandatory | Type | Description |
|-------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |
| template_id | Yes | String | Template ID |

Request Parameters

Table 3-161 Request body parameters

| Parameter | Mandatory | Type | Description |
|-----------|-----------|---------|---------------------|
| status | Yes | Integer | Masking task status |

Response Parameters

Status code: 200

Table 3-162 Response body parameters

| Parameter | Type | Description |
|-----------|--------|-------------------------------------------|
| msg | String | Returned message |
| status | String | Return status, for example, '200', '400'. |

Status code: 400

Table 3-163 Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Start a data masking task.

```
POST /v1/{project_id}/sdg/server/mask/dbs/templates/{template_id}/operation
{
  "status" : 1
}
```

Example Responses

Status code: 200

Request sent

```
{
  "msg" : "xxx",
  "status" : "RESPONSE_SUCCESS"
}
```

Status code: 400

Invalid request

```
{
  "error_code" : "dsc.40000011",
  "error_msg" : "Invalid parameter"
}
```

Status Codes

| Status Code | Description |
|-------------|-----------------|
| 200 | Request sent |
| 400 | Invalid request |

Error Codes

See [Error Codes](#).

3.10 API Call Records

3.10.1 Querying OpenAPI Calls

Function

This API is used to query OpenAPI calling records.

URI

GET /v1/{project_id}/openapi/called-records

Table 3-164 Path Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|-------------|
| project_id | Yes | String | Project ID |

Table 3-165 Query Parameters

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| limit | No | Integer | Page size. The default value is 1000, and the maximum value is 2000. |
| called_url | No | String | URL of the calling records. Example: /v1/{project_id}/sdg/database/watermark/embed |
| start_time | No | Long | Start time of the calling records (Unix timestamp in milliseconds). Example: 0 |
| end_time | No | Long | End time of the calling records (Unix timestamp in milliseconds). Example: 1638515803572 |
| marker | No | String | Marker of the next page. This parameter is not required for the first page. You need to set this parameter to a return value in the first page of the query results for the next page. |

Request Parameters

None

Response Parameters

Status code: 200

Table 3-166 Response body parameters

| Parameter | Type | Description |
|-----------|---------|----------------------------|
| total | Integer | Total number of API calls |
| succeed | Integer | Number of successful calls |
| failed | Integer | Number of failed calls |

| Parameter | Type | Description |
|------------------------|------------------------------------------------------|-------------------------|
| openapi_called_records | Array of OpenApiCalledRecord objects | List of OpenAPI calls |
| next_marker | String | Marker of the next page |

Table 3-167 OpenApiCalledRecord

| Parameter | Type | Description |
|----------------|--------|-----------------------------------------------------|
| user_name | String | Name of the user that called the APIs |
| user_id | String | ID of the user that called the APIs |
| domain_name | String | Domain name of the API calls |
| domain_id | String | Domain IDs of the API calls |
| request_url | String | URL of the API calls |
| request_method | String | HTTP request method |
| response_code | String | HTTP request code |
| fail_reason | String | Failure cause |
| timestamp | Long | Time of the calls (Unix timestamp, in milliseconds) |

Status code: 400**Table 3-168** Response body parameters

| Parameter | Type | Description |
|------------|--------|---------------|
| error_code | String | Error Code |
| error_msg | String | Error Message |

Example Requests

Querying OpenAPI Calls

GET /v1/{project_id}/openapi/called-records

A Appendixes

A.1 Status Codes

| Code | Status | Description |
|------|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 200 | OK | Request successful. |
| 400 | Bad Request | Request failed. Modify the request and then try again. |
| 401 | Unauthorized | Authentication failed. The authentication information provided by the client is incorrect or invalid. |
| 403 | Forbidden | Request rejected. Modify the request and then try again. The server has received and understood the request; yet it refused to respond, because the request is set to deny access. |
| 404 | NotFound | Cannot find the requested resource. Modify the request and then try again. |
| 500 | InternalServerError | Internal error. The server is able to receive the request but it could not understand the request. |

A.2 Error Codes

| Status Code | Error Codes | Error Message | Description | Solution |
|-------------|--------------|---------------------------------------------------------|----------------------------------------------------------|-------------------------------------------------------------------|
| 400 | DSC.00000001 | The project ID is not in the trust list | The project ID is not in the trust list! | Try again. If the operation fails, contact the technical support. |
| 400 | DSC.00000002 | Internal service error | Internal service error! | Try again. If the operation fails, contact the technical support. |
| 400 | DSC.00000003 | Failed to read the upload file | Failed to read the upload file! | Check the file type and size. |
| 400 | DSC.00000004 | Invalid parameter | Invalid parameter! | Check the parameter settings. |
| 400 | DSC.00000005 | Invalid project id | Invalid project id! | Try again. If the operation fails, contact the technical support. |
| 400 | DSC.00000006 | File size exceeds maximum limit | File size exceeds maximum limit! | Check the file size. |
| 400 | DSC.00000007 | File format error | File format error! | Check the file format. |
| 400 | DSC.00000008 | Upload file is empty | Upload file is empty! | Check whether the file is empty. |
| 400 | DSC.00000009 | There are many tasks at present. Please try again later | There are many tasks at present. Please try again later! | Try again. If the operation fails, contact the technical support. |
| 400 | DSC.00000010 | Image watermark file is empty | Image watermark file is empty! | Add a graphic file. |
| 400 | DSC.00000011 | Image watermark file exceeds maximum limit | Image watermark file exceeds maximum limit! | limit the size of the graphic file. |

| Status Code | Error Codes | Error Message | Description | Solution |
|-------------|--------------------|-------------------------------------------|--------------------------------------------|-------------------------------------------------------------------|
| 400 | DSC.00000012 | Invalid project image watermark file type | Invalid project image watermark file type! | Upload a .png or .jpg file. |
| 400 | DSC.00000013 | Invalid Auth | Invalid Auth! | Use a valid token. |
| 400 | DSC.00000014 | Order DSC professional version | Order DSC professional version! | Order DSC professional version first. |
| 400 | mask.20000001 | Unknown error | Unknown error! | Try again. If the operation fails, contact the technical support. |
| 400 | mask.20000002 | Internal service error | Internal service error! | Try again. If the operation fails, contact the technical support. |
| 400 | mask.20000003 | Invalid parameter | Invalid parameter! | Pass a correct parameter. |
| 400 | mask.20000004 | Mask data failed | Mask data failed! | Check the parameter settings. |
| 400 | watermark.10000001 | Unknown error | Unknown error! | Try again. If the operation fails, contact the technical support. |
| 400 | watermark.10000002 | Internal service error | Internal service error! | Try again. If the operation fails, contact the technical support. |
| 400 | watermark.10000003 | Failed to read the upload file | Failed to read the upload file! | Check the file type and size. |
| 400 | watermark.10000004 | Invalid parameter | Invalid parameter! | Check the parameter settings. |
| 400 | watermark.10000005 | Watermark not found | Watermark not found! | Check whether the file contains a watermark. |

| Status Code | Error Codes | Error Message | Description | Solution |
|-------------|-------------------|---------------------------------------------------------------------------------|----------------------------------------------------------------------------------|-------------------------------------------------------------------|
| 400 | watermark.1000006 | File size exceeds maximum limit | File size exceeds maximum limit! | Check the file size. |
| 400 | watermark.1000007 | File format error | File format error! | Check the file format. |
| 400 | watermark.1000008 | Upload file is empty | Upload file is empty! | Check whether the file is empty. |
| 400 | watermark.1000009 | Wrong password | Wrong password! | Check whether the file password is correct. |
| 400 | watermark.1000010 | The ".doc" format file does not support the watermark feature currently | The ".doc" format file does not support the watermark feature currently! | Check the file format. |
| 400 | watermark.1000011 | The ".xls" format file does not support the visible watermark feature currently | The ".xls" format file does not support the visible watermark feature currently! | Check the file format. |
| 400 | watermark.1000012 | The ".ppt" format file does not support the visible watermark feature currently | The ".ppt" format file does not support the visible watermark feature currently! | Check the file format. |
| 400 | watermark.1000013 | No watermark specified | No watermark specified! | Check whether the data watermarking parameters are set. |
| 400 | watermark.1000014 | There are many tasks at present. Please try again later | There are many tasks at present. Please try again later! | Try again. If the operation fails, contact the technical support. |

| Status Code | Error Codes | Error Message | Description | Solution |
|-------------|-------------------|--------------------------------------------------------------------------|---------------------------------------------------------------------------|--------------------------------------------------------------------------|
| 400 | watermark.1000015 | Blind watermark already exists | Blind watermark already exists! | Check whether the file contains an invisible watermark. |
| 400 | watermark.1000016 | There is not enough channel capacity to embed the watermark | There is not enough channel capacity to embed the watermark! | Check the file size. |
| 400 | watermark.1000017 | Image channel capacity is too small to embed watermark | Image channel capacity is too small to embed watermark! | Try again. If the operation fails, contact the technical support. |
| 400 | watermark.1000018 | Currently does not support embedding watermark to image below 512 pixels | Currently does not support embedding watermark to image below 512 pixels! | Please upload image larger than 512 pixels, or contact customer service. |

A.3 Obtaining a Project ID

Obtaining a Project ID by Calling an API

You can obtain the project ID by calling the IAM API used to query project information based on the specified criteria.

The API used to obtain a project ID is GET `https://{Endpoint}/v3/projects`. **{Endpoint}** is the IAM endpoint and can be obtained from the administrator. For details about API authentication, see [Authentication](#).

In the following example, **id** indicates the project ID.

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "xxxxxxx",
      "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"
}
}

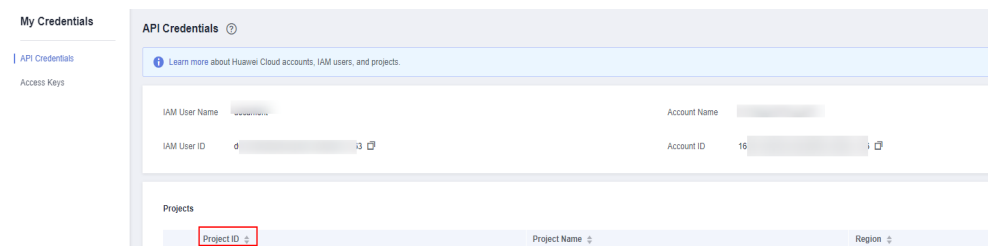
```

Obtaining a Project ID from the Console

A project ID is required for some URLs when an API is called. To obtain a project ID, perform the following operations:

1. Log in to the management console.
2. Click the username and choose **My Credentials** from the drop-down list.
3. On the page, view the project ID in the project list.

Figure A-1 Viewing project IDs



A.4 Configuring a Dynamic Sensitive Data Masking Policy

A.4.1 SHA-256/512

SHA-256/512, message-digest algorithms, are used by DSC to compute a digest from a string in a specified field.

Request Parameters

Table A-1 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|--------|----------------------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, SHA-256 and SHA-512 |
| parameters | Yes | Object | Parameter, which can be ignored |

Example Request

```
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "SHA256",
      "parameters": {}
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
```

A.4.2 AES

Encrypt the character string using the AES encryption algorithm.

AES algorithm configurations: The initial vector (IV) is a 16-byte random number. The encryption mode is GCM. PKCS7-Padding (CMS-Padding) is recommended.

In the encryption and data masking result, the first 16 bytes of an encrypted string is the initialization vector (IV) and the rest is the enciphered text. The ciphertext is in binary format. The ciphertext entered into the masking engine is encoded using Base64 and stored in the database as a character string.

Request Parameters

Table A-2 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, AES |
| parameters | Yes | For details, see Table A-3 . | Parameters for configuring a data masking algorithm |

Table A-3 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|---------------------------------------------------------------------------------------------|
| key | Yes | String | AES algorithm key |
| len | Yes | String | Key group length Currently, only 128 , 192 , and 256 are supported. |

Example Request

```
{
  "mask_strategies": [
```



```
{
  "name": "col1",
  "algorithm": "AES",
  "parameters": {
    "key": "df643533b90b6926c9bff63cc16173db",
    "len": "128"
  }
},
"data": [
  {
    "col1": "test"
  }
]
```

A.4.3 PRESNM

Retain the first n and last m characters and mask the content in the middle part of the specified character string.

Request Parameters

Table A-4 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, PRESNM |
| parameters | Yes | For details, see Table A-5 . | Parameters for configuring a data masking algorithm |

Table A-5 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes | String | Character masking methods are described as follows: <ul style="list-style-type: none"> • CHAR: Character masking • RAND: Random masking |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| method | Yes | String | Masking method If type is set to RAND , this parameter can be set as follows: <ul style="list-style-type: none"> • CHAR: Replace the data to be masked with characters. • DIGITAL: Replace the data to be masked with numbers. • BOTH: Replace the data to be masked with letters and numbers. |
| n | Yes | Int | Retain the first <i>n</i> characters. |
| m | Yes | Int | Retain the last <i>m</i> characters. |

Example Request

```

{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "PRESNM",
      "parameters": {
        "type": "CHAR",
        "n": 1,
        "m": 1,
        "method": "*"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
Or
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "PRESNM",
      "parameters": {
        "type": "RAND",
        "n": 1,
        "m": 1,
        "method": "BOTH"
      }
    }
  ],
  "data": [

```

```
{
  "col1": "test"
}
```

A.4.4 MASKNM

Mask the first n and last m characters and retain the content in the middle part of the specified character string.

Request Parameters

Table A-6 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, MASKNM |
| parameters | Yes | For details, see Table A-7 . | Parameters for configuring a data masking algorithm |

Table A-7 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes | String | Character masking methods are described as follows: <ul style="list-style-type: none"> • CHAR: Character masking • RAND: Random masking |
| method | Yes | String | Masking method If type is set to RAND , this parameter can be set as follows: <ul style="list-style-type: none"> • CHAR: Replace the data to be masked with characters. • DIGITAL: Replace the data to be masked with numbers. • BOTH: Replace the data to be masked with letters and numbers. |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|------|-------------------------------------|
| <i>n</i> | Yes | Int | Mask the first <i>n</i> characters. |
| <i>m</i> | Yes | Int | Mask the last <i>m</i> characters. |

Example Request

```
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "MASKNM",
      "parameters": {
        "type": "CHAR",
        "n": 1,
        "m": 1,
        "method": "*"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
Or
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "MASKNM",
      "parameters": {
        "type": "RAND",
        "n": 1,
        "m": 1,
        "method": "BOTH"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
```

A.4.5 PRESXY

Retain the specified character string from *x* to *y* and mask the rest characters.

Request Parameters

Table A-8 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, PRESXY |
| parameters | Yes | For details, see Table A-9 . | Parameters for configuring a data masking algorithm |

Table A-9 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes | String | Character masking methods are described as follows: <ul style="list-style-type: none"> • CHAR: Character masking • RAND: Random masking |
| method | Yes | String | Masking method If type is set to RAND , this parameter can be set as follows: <ul style="list-style-type: none"> • CHAR: Replace the data to be masked with characters. • DIGITAL: Replace the data to be masked with numbers. • BOTH: Replace the data to be masked with letters and numbers. |
| x | Yes | Int | Start position of retained characters |
| y | Yes | Int | End position of retained characters |

Example Request

```
{
  "mask_strategies": [
```

```

    {
      "name": "col1",
      "algorithm": "PRESXY",
      "parameters": {
        "type": "CHAR",
        "x": 1,
        "y": 1,
        "method": "*"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
Or
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "PRESXY",
      "parameters": {
        "type": "RAND",
        "x": 1,
        "y": 1,
        "method": "CHAR"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}

```

A.4.6 MASKXY

Mask the specified character string from x to y and retain the rest characters.

Request Parameters

Table A-10 Parameter description

| Parameter | Mandatory | Type | Description |
|-----------------------------------------------------|-----------|-----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, MASKXY |
| Parameters for configuring a data masking algorithm | Yes | For details, see Table A-11 . | Parameters for configuring a data masking algorithm |

Table A-11 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes | String | Character masking methods are described as follows: <ul style="list-style-type: none"> • CHAR: Character masking • RAND: Random masking |
| method | Yes | String | Masking method If type is set to RAND , this parameter can be set as follows: <ul style="list-style-type: none"> • CHAR: Replace the data to be masked with characters. • DIGITAL: Replace the data to be masked with numbers. • BOTH: Replace the data to be masked with letters and numbers. |
| x | Yes | Int | Start position of masked characters |
| y | Yes | Int | End position of masked characters |

Example Request

```

{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "MASKXY",
      "parameters": {
        "type": "CHAR",
        "x": 1,
        "y": 1,
        "method": "*"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
Or
{
  "mask_strategies": [

```

```

{
  "name": "col1",
  "algorithm": "MASKXY",
  "parameters": {
    "type": "RAND",
    "x": 1,
    "y": 1,
    "method": "CHAR"
  }
},
"data": [
  {
    "col1": "test"
  }
]
}

```

A.4.7 SYMBOL

Mask the content before or after a special character and retain the rest characters.

Request Parameters

Table A-12 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|-----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, SYMBOL |
| parameters | Yes | For details, see Table A-13 . | Parameters for configuring a data masking algorithm |

Table A-13 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes | String | Character masking methods are described as follows: <ul style="list-style-type: none"> • CHAR: Character masking • RAND: Random masking |

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| method | Yes | String | Masking method If type is set to RAND , this parameter can be set as follows: <ul style="list-style-type: none"> • CHAR: Replace the data to be masked with characters. • DIGITAL: Replace the data to be masked with numbers. • BOTH: Replace the data to be masked with letters and numbers. |
| direction | Yes | Int | Masking direction 0 : Mask the content before a special character. 1 : Mask the content after a special character. |
| symbol | Yes | String | Specified special character |

Example Request

```

{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "SYMBOL",
      "parameters": {
        "type": "CHAR",
        "direction": 1,
        "symbol": "@",
        "method": "x"
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
Or
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "SYMBOL", // Parameter for configuring a data masking algorithm
      "parameters": {
        "type": "RAND",
        "direction": 0,
        "symbol": "@",
        "method": "CHAR"
      }
    }
  ]
}

```

```

    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}

```

A.4.8 KEYWORD

Replace the keyword specified in a character string.

Request Parameters

Table A-14 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|-----------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, KEYWORD |
| parameters | Yes | For details, see Table A-15 . | Parameters for configuring a data masking algorithm |

Table A-15 Parameters for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|---------------------|
| key | Yes | String | Specified keyword |
| target | Yes | String | Value replaced with |

Example Request

```

{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": "KEYWORD",
      "parameters": {
        "key": "Keyword",
        "target": "Target character string"
      }
    }
  ],
  "data": [
    {
      "col1": "Keyword"
    }
  ]
}

```

A.4.9 NUMERIC

Round down a field of the numeric type to the nearest integer.

Request Parameters

Table A-16 Parameter description

| Parameter | Mandatory | Type | Description |
|------------|-----------|---------------------------------------------|-----------------------------------------------------|
| algorithm | Yes | String | Algorithm type, for example, NUMERIC |
| parameters | Yes | For details, see Table A-17 | Parameters for configuring a data masking algorithm |

Table A-17 Parameter for configuring a data masking algorithm

| Parameter | Mandatory | Type | Description |
|-----------|-----------|--------|------------------------------------------------------------------------------------------------------------------|
| value | Yes | Double | Value of the numeric field to rounded down, which can only be an integer or a decimal and must be greater than 0 |

Example Request

```
{
  "mask_strategies": [
    {
      "name": "col1",
      "algorithm": " NUMERIC",
      "parameters": {
        "value": 0.05
      }
    }
  ],
  "data": [
    {
      "col1": "test"
    }
  ]
}
```

B Change History

| Date | Description |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2024-03-30 | <p>This issue is the second official release, which incorporates the following change:</p> <p>Deleted:</p> <ul style="list-style-type: none">The API for querying database lineage graph databaseThe API for querying the database lineage graph table in pagesThe API for querying database lineage graph fieldsThe API for querying OBS lineage graph at the bucket levelThe API for querying OBS lineage graph files in pages <p>Added:</p> <ul style="list-style-type: none">The API for deleting scan tasks |
| 2023-11-30 | This issue is the first official release. |