SSL Certificate Manager

API Reference

Issue 10

Date 2021-05-07

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

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

Contents

1 Before You Start

1.1 Overview

Welcome to SSL Certificate Manager (SCM) API Reference. SCM provides you with a one-stop management service for SSL certificates throughout their lifecycles. Jointly developed by HUAWEI CLOUD and globally well-known digital certificate agencies, SCM implements trusted identity authentication and secure data transmission for websites.

You can use the APIs provided in this document to manage your certificates, such as requesting a certificate, querying the certificate list, and deleting a certificate. For details about all supported operations, see **API Overview**.

Before calling SCM APIs, ensure that you have understood the concepts related to SCM. For more information, see **What Is SSL Certificate Manager?**

1.2 API Calling

SCM supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see **Calling APIs**.

1.3 Endpoints

An endpoint is the **request address** for calling an API. SCM is a global service deployed for all physical regions. **Table 1-1** lists the endpoints of SCM. You can obtain SCM endpoints at **Regions and Endpoints**.

Table 1-1 SCM endpoints

Region	Endpoint Region	Endpoint	Protocol Type
All	All	scm.cn-north-4.myhuaweicloud.com	HTTPS

Region	Endpoint Region	Endpoint	Protocol Type
All	ALL	scm.ap-southeast-1.myhuaweicloud.com	HTTPS

1.4 Constraints

For more constraints, see the API description.

1.5 Concepts

- Account

  An account is created upon successful registration with HUAWEI CLOUD. 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 used to perform routine management. For security purposes, create IAM users under the account 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).

  An IAM user can view the account ID and user ID on the **My Credentials** page of the console. 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 tenants. A dedicated region provides services of the same type only or for specific tenants.

  For details, see **Region and AZ**.

- Availability Zone (AZ)

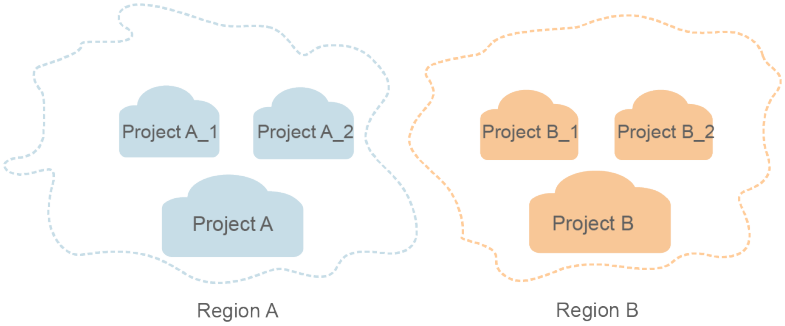
  An AZ comprises one or multiple physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Compute, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers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

- Project

  Projects group and isolate resources (including compute, storage, and network resources) across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. 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



- Enterprise project

  Enterprise projects group and manage resources across regions. Resources in enterprise projects are logically isolated from each other. An enterprise project can contain resources in multiple regions, and resources can be directly transferred between enterprise projects.

  For details about how to obtain enterprise project IDs and features, see **Enterprise Management User Guide**.

1.6 Selecting an API Type

For SSH key pairs, V2.1 and V2 API Types are available. It is recommended that you choose V2.1, which can better meet your demands.

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 **Regions and Endpoints**.

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

- **resource-path**:

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

- **query-string**:

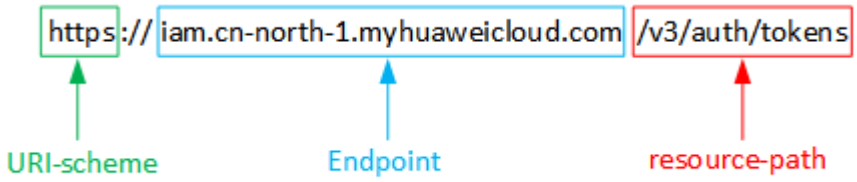
  Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of "Parameter name=Parameter value". For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

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

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

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

Figure 2-1 Example URI



NOTE

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

Request Methods

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

- **GET**: requests the server to return specified resources.
- **PUT**: requests the server to update specified resources.
- **POST**: requests the server to add resources or perform special operations.
- **DELETE**: requests the server to delete specified resources, for example, an object.
- **HEAD**: same as GET except that the server must return only the response header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource 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.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

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

Common request header fields are as follows:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. 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.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body

The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header.

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 ***xxxxxxxxxxxxxxxxx*** to the project name, such as **cn-north-1**. You can learn more information about projects from **Regions and Endpoints**. Check the value of the **Region** column.

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

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

```
    }
  }
}
```

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

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:


```
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. For details about how to sign requests and use the signing SDK, see **API Signature Guide**.

NOTICE

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

2.3 Returned Values

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 digits ranging from *1xx* to *5xx*. It indicates the status of a response. For more information, see **Status Codes**.

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

Response Header

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

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

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

(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 **obtain a user token**. 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": "xxxxxxxx",
......
```

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 Overview

By using the APIs provided by SCM, you can use all functions of SCM.

API	Description
Querying the certification list	Query the certificate list.
Importing an external certificate	Import a certificate to SCM.
Obtaining the certificate details	Query details of a certificate.
Deleting a certificate	Delete a certificate.
Exporting a certificate	Export a certificate.
Pushing a certificate	Push a certificate to another HUAWEI CLOUD service.

4 API Description

4.1 Querying the Certificate List

Function

This API is used to query the certificate list by certificate name or bound domain name.

URI

GET /v3/scm/certificates

Table 4-1 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of certificate records displayed on each page. The values can be: • 10: 10 certificate records can be displayed on each page. • 20: 20 certificate records can be displayed on each page. • 50: 50 certificate records can be displayed on each page. Minimum: **10** Maximum: **50** Default: **10**

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset. The value ranges from 1 to 30. Minimum: **0** Maximum: **30** Default: **0**
sort_dir	No	String	Sorting method. Sorting is performed based on the sorting parameter sort_key. The value can be: • ASC: Ascending order • DESC: descending order. Default: **DESC** Minimum: **0** Maximum: **32**
sort_key	No	String	Parameter by which the certificates are sorted out. The value can be: • certExpiredTime: certificate expiration time. • certStatus: certificate status. • certUpdateTime: certificate update time. Default: **certUpdateTime** Minimum: **0** Maximum: **64**

Parameter	Mandatory	Type	Description
status	No	String	Certificate status. The value can be: • ALL: All certificate statuses. • PAID: The certificate has been paid and needs to be applied for from the CA. • ISSUED: The certificate has been issued. • CHECKING: The certificate application is being reviewed. • CANCELCHECKING: The certificate application cancellation is being reviewed. • UNPASSED: The certificate application fails. • EXPIRED: The certificate has expired. • REVOKING: The certificate revocation application is being reviewed. • REVOKED: The certificate has been revoked. • UPLOAD: The certificate is being hosted. • CHECKING_ORG: the organization verification is to be completed. • ISSUING: The certificate is to be issued. • SUPPLEMENTCHECKING: Additional domain names to be added for a multi-domain certificate are being reviewed. Default: **ALL** Minimum: **0** Maximum: **64**

Request Parameters

Table 4-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: **32** Maximum: **2097152**

Response Parameters

Status code: 200

Table 4-3 Response body parameters

Parameter	Type	Description
certificates	Array of **CertificateDetail** objects	For details, see data structure description of the CertificateDetail field.
total_count	Integer	Number of certificates. Minimum: **0** Maximum: **10000**

Table 4-4 CertificateDetail

Parameter	Type	Description
id	String	Certificate ID. Minimum: **16** Maximum: **16**
name	String	Certificate name. Maximum: **255** Minimum: **1**
domain	String	Domain name bound to a certificate. Minimum: **1** Maximum: **255**

Parameter	Type	Description
type	String	Certificate type. The value can be: DV_SSL_CERT, DV_SSL_CERT_BASIC, EV_SSL_CERT, EV_SSL_CERT_PRO, OV_SSL_CERT, or OV_SSL_CERT_PRO Minimum: **1** Maximum: **128**
brand	String	Certificate authority. The value can be GLOBALSIGN, SYMANTEC, GEOTRUST or CFCA. Minimum: **1** Maximum: **255**
expire_time	String	Certificate expiration time. Minimum: **1** Maximum: **32**
domain_type	String	Domain type. The value can be: • SINGLE_DOMAIN: single domain names • WILDCARD: wildcard domains • MULTI_DOMAIN: multiple domain names Minimum: **1** Maximum: **128**
validity_period	Integer	Certificate validity period, in months. Minimum: **12** Maximum: **12**

Parameter	Type	Description
status	String	Certificate status. The value can be: • PAID: The certificate has been paid, and needs to be applied for from the CA. • ISSUED: The certificate has been issued. • CHECKING: The certificate application is being reviewed. • CANCELCHECKING: The certificate application cancellation is being reviewed. • UNPASSED: The certificate application fails. • EXPIRED: The certificate has expired. - REVOKING: The certificate revocation application is being reviewed. • CANCLEREVOKING: The cancellation on certificate revocation is being reviewed. • REVOKED: The certificate has been revoked. • UPLOAD: The certificate is being hosted. • SUPPLEMENTCHECKING: Additional domain names to be added for a multi-domain certificate are being reviewed. • CANCELSUPPLEMENTING: The cancellation on additional domain names to be added is being reviewed. Minimum: **0** Maximum: **64**
domain_count	Integer	Number of domain names can be bound to a certificate. Minimum: **1** Maximum: **100**
wildcard_count	Integer	Number of wildcard domain names can be bound to a certificate. Minimum: **0** Maximum: **100**
description	String	Certificate description Minimum: **0** Maximum: **255**

Status code: 401

Table 4-5 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-6 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-7 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 200

Normal return.

```
{
  "certificates" : [ {
    "id" : "scs1554192131150",
    "name" : "test",
    "domain" : "www.zx.com",
    "type" : "OV_SSL_CERT",
    "brand" : "GEOTRUST",
    "expire_time" : "2021-05-27 16:46:25.0",
    "domain_type" : "MULTI_DOMAIN",
    "validity_period" : 12,
    "status" : "ISSUED",
    "domain_count" : 2,
    "wildcard_count" : 0,
    "description" : null
  } ],
  "total_count" : 1
}
```

Status code: 401

Authentication failed.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 403

Access denied.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 500

Failed to complete the request because of an internal server error.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status Codes

Status Code	Description
200	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

4.2 Importing a Certificate

Function

This API is used to import a certificate to SCM.

URI

POST /v3/scm/certificates/import

Request Parameters

Table 4-8 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: **32** Maximum: **2097152**

Table 4-9 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Certificate name. The value contains 0 to 63 characters. Minimum: **0** Maximum: **255**
certificate	Yes	String	Certificate content. Use the escape character \n or \r\n to replace carriage return and line feed characters. Minimum: **0** Maximum: **4096**

Parameter	Mandatory	Type	Description
certificate_chain	Yes	String	Certificate chain. Use the escape character \n or \r\n to replace carriage return and line feed characters. Minimum: **0** Maximum: **8192**
private_key	Yes	String	Private key of a certificate. The private key protected by password cannot be uploaded. The carriage return character must be replaced with the escape character \n or \r\n. Minimum: **0** Maximum: **4096**

Response Parameters

Status code: 200

Table 4-10 Response body parameters

Parameter	Type	Description
certificate_id	String	Certificate ID. Minimum: **16** Maximum: **16**

Status code: 401

Table 4-11 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-12 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-13 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 200

Normal return.

```
{
  "certificate_id" : "scs1600313391074"
}
```

Status code: 401

Authentication failed.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 403

Access denied.


```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 500

Failed to complete the request because of an internal server error.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status Codes

Status Code	Description
200	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

4.3 Obtaining Details of a Certificate

Function

This API is used to query details about a certificate.

URI

GET /v3/scm/certificates/{certificate_id}

Table 4-14 Path parameters

Parameter	Mandatory	Type	Description
certificate_id	Yes	String	Certificate ID. Minimum: **16** Maximum: **16**

Request Parameters

Table 4-15 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: **32** Maximum: **2097152**

Response Parameters

Status code: 200

Table 4-16 Response body parameters

Parameter	Type	Description
id	String	Certificate ID. Minimum: **16** Maximum: **16**

Parameter	Type	Description
status	String	Certificate status. The value can be: • PAID: The certificate has been paid, and needs to be applied for from the CA. • ISSUED: The certificate has been issued. • CHECKING: The certificate application is being reviewed. • CANCELCHECKING: The certificate application cancellation is being reviewed. • UNPASSED: The certificate application fails. • EXPIRED: The certificate has expired. • REVOKING: The certificate revocation application is being reviewed. • REVOKED: The certificate has been revoked. • UPLOAD: The certificate is being hosted. • SUPPLEMENTCHECKING: Additional domain names to be added for a multi-domain certificate are being reviewed. • CANCELSUPPLEMENTING: The cancellation on additional domain names to be added is being reviewed. Minimum: **0** Maximum: **32**
order_id	String	Order ID Minimum: **36** Maximum: **36**
name	String	Certificate name. Minimum: **0** Maximum: **255**
type	String	Certificate type. The value can be: DV_SSL_CERT, DV_SSL_CERT_BASIC, EV_SSL_CERT, EV_SSL_CERT_PRO, OV_SSL_CERT, or OV_SSL_CERT_PRO Minimum: **0** Maximum: **32**
brand	String	Certificate authority. The value can be: GLOBALSIGN, SYMANTEC, GEOTRUST or CFCA. Minimum: **0** Maximum: **32**

Parameter	Type	Description
push_support	String	Whether a certificate can be pushed. Minimum: **0** Maximum: **32**
revoke_reason	String	Reason for certificate revocation. Minimum: **0** Maximum: **255**
signature_algrithm	String	Signature algorithm. Minimum: **0** Maximum: **64**
issue_time	String	Certificate issuance time. If no valid value is obtained, this parameter is left blank. Minimum: **0** Maximum: **32**
not_before	String	Time when the certificate takes effect. If no valid value is obtained, this parameter is left blank. Minimum: **0** Maximum: **32**
not_after	String	Time when the certificate becomes invalid. If no valid value is obtained, this parameter is left blank. Minimum: **0** Maximum: **32**
validity_period	Integer	Certificate validity period, in months. Minimum: **12** Maximum: **12**
validation_method	String	Domain ownership verification method. The value can be DNS, FILE, or EMAIL. Minimum: **0** Maximum: **32**
domain_type	String	Domain type. The value can be: ● SINGLE_DOMAIN: single domain names ● WILDCARD: wildcard domains ● MULTI_DOMAIN: multiple domain names Minimum: **0** Maximum: **32**

Parameter	Type	Description
domain	String	Domain name bound to a certificate. Minimum: **0** Maximum: **255**
sans	String	Additional domain name associated with the certificate Minimum: **0** Maximum: **4096**
domain_count	Integer	Number of domain names can be bound to a certificate. Minimum: **1** Maximum: **100**
wildcard_count	Integer	Number of additional domain names can be bound to a certificate. Minimum: **0** Maximum: **99**
authentification	Array of **Authentification** objects	Domain ownership verification information. For details, see data structure of the Authentification field.

Table 4-17 Authentification

Parameter	Type	Description
record_name	String	Name of a domain ownership verification value. Minimum: **0** Maximum: **255**
record_type	String	Type of the domain name verification value. Minimum: **0** Maximum: **255**
record_value	String	Domain verification value. Minimum: **0** Maximum: **255**
domain	String	Domain name mapping to the verification value Minimum: **0** Maximum: **255**

Status code: 401

Table 4-18 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-19 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-20 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 200

Normal return.


```
{
  "id" : "scs1590053258704",
  "order_id" : "CS20052117270N7V9",
  "name" : "scm-testing",
  "type" : "DV_SSL_CERT",
  "brand" : "SYMANTEC",
  "push_support" : "OFF",
  "revoke_reason" : null,
  "status" : "CHECKING_DOMAIN",
  "signature_algrithm" : null,
  "issue_time" : null,
  "not_before" : null,
  "not_after" : null,
  "validity_period" : 12,
  "validation_method" : "DNS",
  "domain_type" : "SINGLE_DOMAIN",
  "domain" : "hosting-******.hwcloudtest.cn",
  "sans" : null,
  "domain_count" : 1,
  "wildcard_count" : 0,
  "authentification" : [ {
    "record_name" : "_dnsauth.hosting-****.hwcloudtest.cn",
    "record_type" : "TXT",
    "record_value" : "201801040000001ytm4q**********cd8p7eg9ktlwfsord",
    "domain" : "hosting-*****.hwcloudtest.cn"
  } ]
}
```


Status code: 401

Authentication failed.


```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```


Status code: 403

Access denied.


```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```


Status code: 500

Failed to complete the request because of an internal server error.


```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```


Status Codes

Status Code	Description
200	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

4.4 Deleting a Certificate

Function

This API is used to delete a certificate from HUAWEI CLOUD.

URI

DELETE /v3/scm/certificates/{certificate_id}

Table 4-21 Path parameters

Parameter	Mandatory	Type	Description
certificate_id	Yes	String	Certificate ID. Minimum: **16** Maximum: **16**

Request Parameters

None

Response Parameters

Status code: 401

Table 4-22 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-23 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-24 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 401

Authentication failed.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 403

Access denied.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 500

Failed to complete the request because of an internal server error.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status Codes

Status Code	Description
204	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

4.5 Exporting a Certificate

Function

This API is used to export a certificate.

URI

POST /v3/scm/certificates/{certificate_id}/export

Table 4-25 Path parameters

Parameter	Mandatory	Type	Description
certificate_id	Yes	String	Certificate ID. Minimum: **16** Maximum: **16**

Request Parameters

Table 4-26 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: **32** Maximum: **2097152**

Response Parameters

Status code: 200

Table 4-27 Response body parameters

Parameter	Type	Description
certificate	String	Certificate content Minimum: **1** Maximum: **4096**
certificate_chain	String	Certificate chain Minimum: **1** Maximum: **8192**
private_key	String	Private key of a certificate Minimum: **1** Maximum: **4096**

Status code: 401

Table 4-28 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-29 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-30 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 200

Normal return.

```
{
  "certificate" : "-----BEGIN CERTIFICATE-----*******-----END CERTIFICATE-----",
  "certificate_chain" : "-----BEGIN CERTIFICATE-----*******-----END CERTIFICATE-----",
  "private_key" : "-----BEGIN RSA PRIVATE KEY-----******-----END RSA PRIVATE KEY-----"
}
```

Status code: 401

Authentication failed.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 403

Access denied.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 500

Failed to complete the request because of an internal server error.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status Codes

Status Code	Description
200	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

4.6 Pushing a Certificate

Function

This API is used to push an SSL certificate to another HUAWEI CLOUD service, such as Elastic Load Balance (ELB), Web Application Firewall (WAF), and Content Delivery Network (CDN).

URI

POST /v3/scm/certificates/{certificate_id}/push

Table 4-31 Path parameters

Parameter	Mandatory	Type	Description
certificate_id	Yes	String	Certificate ID. Minimum: **16** Maximum: **16**

Request Parameters

Table 4-32 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: **32** Maximum: **2097152**

Table 4-33 Request body parameters

Parameter	Mandatory	Type	Description
target_project	Yes	String	Region where the service you want to push a certificate to is deployed. Minimum: **1** Maximum: **255**

Parameter	Mandatory	Type	Description
target_service	Yes	String	Service to which the certificate is pushed. Currently, certificates can only be pushed to CDN, WAF, and ELB. Minimum: **1** Maximum: **64**

Response Parameters

Status code: 401

Table 4-34 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 403

Table 4-35 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Status code: 500

Table 4-36 Response body parameters

Parameter	Type	Description
error_code	String	Error code returned for an error request. Minimum: **3** Maximum: **36**
error_msg	String	Error information returned for an error request. Minimum: **0** Maximum: **1024**

Example Requests

None

Example Responses

Status code: 401

Authentication failed.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 403

Access denied.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status code: 500

Failed to complete the request because of an internal server error.

```
{
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
}
```

Status Codes

Status Code	Description
204	Normal return.
401	Authentication failed.
403	Access denied.
404	Access page not found.

Status Code	Description
500	Failed to complete the request because of an internal server error.

Error Codes

See **Error Codes**.

5 Examples

5.1 Example 1: Deleting an Expired Certificate

Scenario

This topic describes how to delete an expired certificate using the SCM API.

NOTE

- Deleting a certificate only deletes it from HUAWEI CLOUD. The certificate will still be valid and trusted by web browsers.
- After you submit a certificate deletion application, you cannot cancel it. Exercise caution when performing this operation.

Involved APIs

- **Querying the certificate list**: Query all certificates under your current account.
- **Deleting a certificate**: Delete a specific expired certificate.

Procedure

Step 1 Query the certificate list.

- API information

  URI format: GET /v3/scm/certificates

  For details, see **Querying the Certificate List**.

- Example request

  GET: https://*{endpoint}*/v3/scm/certificates

  Obtain *{endpoint}* from **Regions and Endpoints**.

  Body:

```
{
   "limit": "2",
   "offset": "0"
}
```


- Example response

```
{
 "certificates" : [ {
  "id" : "scs1554192131150",
  "name" : "test",
  "domain" : "www.zx.com",
  "type" : "OV_SSL_CERT",
  "brand" : "GEOTRUST",
  "expire_time" : "2021-05-27 16:46:25.0",
  "domain_type" : "MULTI_DOMAIN",
  "validity_period" : 12,
  "status" : "ISSUED",
  "domain_count" : 2,
  "wildcard_count" : 0,
  "description" : null
 } ],
 "total_count" : 1
}
```

Step 2 Delete a certificate.

- API information

  URI format: DELETE /v3/scm/certificates/{certificate_id}

  For details, see **Deleting a Certificate**.

- Example request

  DELETE: https://*{endpoint}*/v3/scm/certificates/scs1554192131150

  Obtain *{endpoint}* from **Regions and Endpoints**.

  Body:

```
{
 certificate_id:scs1554192131150
}
```

- Example response

```
{ }
```

  or

```
{
"error_code" : "SCM.XXX",
"error_msg" : "XXX"
}
```

----End

5.2 Example 2: Pushing Your External Certificate to Other Services

Scenario

This topic describes how to import your SSL certificates purchased on other platforms to SCM and push such certificates to your cloud services on HUAWEI CLOUD, such as WAF, ELB, and CDN, to keep your workloads secure.

Involved APIs

- **Importing a certificate**: Import an SSL certificate not purchased on SCM into SCM for centralized management.
- **Querying the certificate list**: Query all certificates under your current account and obtain the ID of the certificate to be pushed.

- **Pushing a certificate**: Push an SSL certificate imported into SCM to other HUAWEI CLOUD services.

Procedure

Step 1 Import the certificate.

- API information

  URI format: POST /v3/scm/certificates/import

  For details, see **Importing a Certificate**
- Example request

  POST: https://*{endpoint}*/v3/scm/certificates/import

  Obtain *{endpoint}* from **Regions and Endpoints**.
- Example response

```
{
  "certificate_id" : "scs1554192131150"
}
```

Step 2 Query the certificate list.

- API information

  URI format: GET /v3/scm/certificates

  For details, see **Querying the Certificate List**.
- Example request

  GET: https://*{endpoint}*/v3/scm/certificates

  Obtain *{endpoint}* from **Regions and Endpoints**.

  Body:

```
{
   "limit": "2",
   "offset": "0"
}
```

- Example response

```
{
  "certificates" : [ {
    "id" : "scs1554192131150",
    "name" : "test",
    "domain" : "www.zx.com",
    "type" : "OV_SSL_CERT",
    "brand" : "GEOTRUST",
    "expire_time" : "2021-05-27 16:46:25.0",
    "domain_type" : "MULTI_DOMAIN",
    "validity_period" : 12,
    "status" : "ISSUED",
    "domain_count" : 2,
    "wildcard_count" : 0,
    "description" : null
  } ],
  "total_count" : 1
}
```

Step 3 Push a certificate.

- API information

  URI format: POST /v3/scm/certificates/{certificate_id}/push

  For details, see **Pushing a Certificate**
- Example request

POST: https://*{endpoint}*/v3/scm/certificates/scs1554192131150/push

Obtain *{endpoint}* from **Regions and Endpoints**.

- Example response

```
{
 }
```

or

```
{
"error_code" : "SCM.XXX",
"error_msg" : "XXX"
}
```

----End

5.3 Example 3: Pushing Your Certificate Purchased in SCM to Other Services

Scenario

This topic describes how to push a certificate purchased and issued in SCM to WAF, ELB, and CDN on HUAWEI CLOUD to keep your workloads secure.

Involved APIs

- **Querying the certificate list**: Query all certificates under your current account and obtain the ID of the certificate to be pushed.
- **Pushing a certificate**: Push an SSL certificate purchased in SCM to other HUAWEI CLOUD services.

Procedure

Step 1 Query the certificate list.

- API information

  URI format: GET /v3/scm/certificates

  For details, see **Querying the Certificate List**.

- Example request

  GET: https://*{endpoint}*/v3/scm/certificates

  Obtain *{endpoint}* from **Regions and Endpoints**.

  Body:

```
{
   "limit": "2",
   "offset": "0"
}
```

- Example response

```
{
 "certificates" : [ {
   "id" : "scs1554192131150",
   "name" : "test",
   "domain" : "www.zx.com",
   "type" : "OV_SSL_CERT",
   "brand" : "GEOTRUST",
   "expire_time" : "2021-05-27 16:46:25.0",
   "domain_type" : "MULTI_DOMAIN",
```



```
  "validity_period" : 12,
  "status" : "ISSUED",
  "domain_count" : 2,
  "wildcard_count" : 0,
  "description" : null
 } ],
 "total_count" : 1
}
```

Step 2 Push a certificate.

- API information

  URI format: POST /v3/scm/certificates/{certificate_id}/push

  For details, see **Pushing a Certificate**

- Example request

  POST: https://*{endpoint}*/v3/scm/certificates/scs1554192131150/push

  Obtain *{endpoint}* from **Regions and Endpoints**.

- Example response

  ```
  {
   }
  ```

  or

  ```
  {
  "error_code" : "SCM.XXX",
  "error_msg" : "XXX"
  }
  ```

----End

6 Historical APIs

6.1 Purchasing an SSL Certificate

Function

This API is used to purchase an SSL certificate.

NOTE

The request parameter **agree_privacy_protection** must be set to **true**. Otherwise, the certificate purchase application cannot be submitted.

URI

- URI format

  POST /v2/{project_id}/scm/cert/purchase

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
cert_brand	Yes	String	Certificate brand. For example: GLOBALSIGN

Parameter	Mandatory	Type	Description
cert_type	Yes	String	Certificate type. Options: • **OV_SSL_CERT**: Organization Validation (OV) SSL certificate. • **EV_SSL_CERT**: Extended Validation (EV) SSL certificate.
domain_type	Yes	String	Domain name type. Options: • **SINGLE_DOMAIN**: single-domain name type. • **MULTI_DOMAIN**: multi-domain name type. • **WILDCARD**: wildcard domain name type.
effective_time	Yes	Integer	Certificate validity period, in years. Options: • **1**: Purchase a certificate with a validity period of one year. • **2**: Purchase a certificate with a validity period of two years.
domain_numbers	Yes	Integer	Number of domain names. • If **domain_type** is set to **SINGLE_DOMAIN** or **WILDCARD**, the value of **domain_numbers** is **1**. • If **domain_type** is set to **MULTI_DOMAIN**, the value of **domain_numbers** ranges from 2 to 100.
order_number	Yes	Integer	Number of purchased certificates. Value range: 1-1000.

Parameter	Mandatory	Type	Description
agree_privacy_protection	Yes	Boolean	Whether to agree with the privacy statement. • **true**: Agree with the privacy statement. • **false**: Disagree with the privacy statement. You can purchase a certificate only when this parameter is set to **true**.

Response

Response parameters

Parameter	Mandatory	Type	Description
order_id	Yes	String	Order ID.
cert	Yes	Array of cert objects	Certificate list. For details, see **Table 6-1**.

Table 6-1 cert

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.

Example

The following describes how to purchase a multi-domain OV certificate issued by GlobalSign. Assume that the domain quantity is 5, and the validity period is one year.

- Example request

```
{
 "cert_brand":"GLOBALSIGN",
 "cert_type":"OV_SSL_CERT ",
 "domain_type":"MULTI_DOMAIN",
 "effective_time": 1,
 "domain_numbers": 5,
 "order_number": 1,
 "agree_privacy_protection":true,
 }
```

- Example response

```
{
"order_id": "CS1803192259ROA8U"
"cert": [{
        "cert_id": "scs1481110651012",
```



```
    }]
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-2 lists the normal status code returned by the API.

Table 6-2 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.2 Querying the Certificate List

Function

This API is used to query the certificate list based on a certificate name or bound domain name.

NOTICE

This API will be discarded. You are advised to query certificates by referring to **Querying the Certificate Lists**.

URI

- URI format

  GET /v2/{project_id}/scm/certlist?order_status=&content=&sort_key=&sort_dir=&limit=&offset=

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
order_status	No	String	Certificate status. Options: • **PAID**: The certificate has been paid. • **ISSUED**: The certificate has been issued. • **CHECKING**: The certificate application is being reviewed. • **CANCELCHECKING**: The certificate application cancellation is being reviewed. • **UNPASSED**: The certificate application fails. • **EXPIRED**: The certificate has expired. • **REVOKING**: The certificate revocation application is being reviewed. • **REVOKED**: The certificate has been revoked. • **UPLOAD**: The certificate is being hosted. • **SUPPLEMENTCHECK-ING**: Additional domain names to be added for a multi-domain certificate is being reviewed. • **CANCELSUPPLEMENT-ING**: The cancellation on additional domain names to be added is being reviewed.
content	No	String	Keyword for search.
sort_key	No	String	Sorting criterion. Options: • **certExpiredTime**: certificate expiration time. • **certStatus**: certificate status. • **certUpdateTime**: certificate update time.

Parameter	Mandatory	Type	Description
sort_dir	No	String	Sorting method. Sorting is performed based on the sorting parameter **sort_key**. Options: • **ASC**: ascending order. • **DESC**: descending order.
limit	No	Integer	Maximum number of pieces of certificate information to be displayed on each page. Options: • **10**: Each page displays up to 10 pieces of certificate information. • **20**: Each page displays up to 20 pieces of certificate information. • **50**: Each page displays up to 50 pieces of certificate information.
offset	No	Integer	Offset. Value range: 1-30.

Response

Response parameters

Parameter	Mandatory	Type	Description
total	Yes	Integer	Number of certificates in a list.
free_remain	Yes	Integer	Remaining quota of the free test certificate.
order_list	Yes	Array of order_list objects	Certificate list. For details, see **Table 6-3**.

Table 6-3 order_list

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.
cert_name	Yes	String	Certificate name.
domain	Yes	String	Bound domain name.

Parameter	Mandatory	Type	Description
cert_type	Yes	String	Certificate type.
cert_brand	Yes	String	Certificate brand.
domain_type	Yes	String	Domain name type.
purchase_period	Yes	Integer	Validity period.
expired_time	Yes	String	Certificate expiration time.
order_status	Yes	String	Certificate status.
domain_num	Yes	Integer	Number of domain names.
wildcard_number	Yes	Integer	Number of wildcard domain names.
cert_des	Yes	String	Certificate description.

Example

- Example request

  None

- Example response

```
{
"total": 1,
"free_remain":"19",
"order_list": [{
"cert_id": "scs1481110651012",
"cert_name": "scs-0001",
"domain": "*.example.com",
"cert_type": "GE00V01",
"cert_brand":"GLOBALSIGN",
"domain_type":" SINGLE_DOMAIN ",
"purchase_period":1,
"expired_time":"15051501510501",
"order_state":"completed ",
"domain_num":10,
"wildcard_number":2,
"cert_des":"**********"
}]
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-4 lists the normal status code returned by the API.

Table 6-4 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.3 Querying Details of a Certificate

Function

This API is used to query details of a certificate.

NOTICE

This API will be discarded. You are advised to obtain certificate details by referring to **Obtaining Details of a Certificate**.

URI

- URI format
  GET /v2/{project_id}/scm/cert/{cert_id}
- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.
order_id	Yes	String	Order ID.
cert_name	Yes	String	Certificate name.

Parameter	Mandatory	Type	Description
cert_type	Yes	String	Certificate type. Example: OV
cert_brand	Yes	String	Certificate brand. Example: GLOBALSIGN
domain_type	Yes	String	Domain name type. Example: MUILT_DOMAIN
domain_name	Yes	String	Domain name bound to a certificate. Example: funnyzx.com;abc.com
domain_number	Yes	Integer	Number of domains. Example: 3
cert_describe	Yes	String	Certificate description.
push_support	Yes	String	Whether a certificate can be pushed.
revoke_reason	Yes	String	Reason for certificate revocation.
domain_name	Yes	String	Domain name bound to a certificate. Multiple domain names are separated by semicolons (;). Example: www.example.com;www.example1.com;www.example2.com
company_name	Yes	String	Company name.
company_province	Yes	String	State or region where a company is located.
company_city	Yes	String	City where a company is located.
applicant_name	Yes	String	Name of a company contact.
applicant_phone	Yes	String	Phone number of a company contact.
applicant_email	Yes	String	Email of a company contact.

Parameter	Mandatory	Type	Description
contact_name	Yes	String	Name of a technical contact.
contact_phone	Yes	String	Phone number of a technical contact.
contact_email	Yes	String	Email of a technical contact.
status	Yes	String	Certificate status.
encrypt_type	Yes	String	Signature encryption algorithm.
country	Yes	String	Country code.
organization_unit	Yes	String	Company department.
DNS_push_status	Yes	String	DNS push status • **ON**: indicates that the push is successful. • **OFF**: indicates that the push fails. • NONE: indicates that the push function is not enabled.
auth	Yes	Array of auth objects	Certificate authentication status. For details, see **Table 6-5**.

Table 6-5 auth

Parameter	Mandatory	Type	Description
method	Yes	String	Authentication method.
status	Yes	String	Certificate authentication status.
domain_name	Yes	String	Domain name for DNS authentication.
host_record	Yes	String	Host record of DNS authentication.
record_type	Yes	String	Record type of DNS authentication.
record	Yes	String	Record value of DNS authentication.

Example

- Example request

  None

- Example response

```
{
"cert_id": "scs1481110651012",
"order_id ": "CS1803192259ROA8U",
"cert_name": "test",
"cert_type": "OV",
"cert_brand": "GEOTRUST",
"domain_type": "MUILT_DOMAIN",
"domain_name": "funnyzx.com;abc.com",
"domain_number": 3,
"cert_describe": "XXXXXXXXX",
"push_support": "on",
"revoke_reason":"xxxxxxxxxxx",
"domain_name": " www.test.com;*.example1.com;*.example2.com",
"company_name": "Huawei Technologies Co., Ltd.",
"company_province": "Guangdong",
"company_city": "Shenzhen",
"applicant_name": "Tom",
"applicant_phone": "13087654321",
"applicant_email": "example@xx.com",
"contact_name": "Jacky",
"contact_phone": "13087654321",
"contact_email": "example@xx.com",
"status": "PAID",
"encrypt_type": "SHA256withRSA2048",
"country": "CN",
"organization_unit": "unit",
"DNS_push_status": "ON",
"auth": [{
"method": "DNS",
"status": " checking ",
"domain_name": "www.test.com",
"host_record": "dnsauth",
"record_type": "TXT",
"record": "201803272148qwedginciog08"
 }]
}
```

or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
 }
```

Status Codes

Table 6-6 lists the normal status code returned by the API.

Table 6-6 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.4 Modifying a Certificate

Function

This API is used to change the name or description of a certificate.

URI

- URI format

  PUT /v2/{project_id}/scm/cert/{cert_id}

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
modify_key	Yes	String	Change key. The value can be **CERT_NAME** or **DESCRIPTION**. • **CERT_NAME**: indicates the name of a certificate to be modified. • **DESCRIPTION**: indicates the description of a certificate to be modified.
modify_value	Yes	String	Modification details. • If the change key is **CERT_NAME**, the value can contain only digits, letters, and hyphens (-). The value is a string of 0 to 63 characters and cannot be null. • When the change key is **DESCRIPTION**, the value is a string of 0 to 255 characters and can be null.

Response

Response parameters

Parameter	Mandatory	Type	Description
response_info	Yes	String	Request result.

Examples

The following describes how to change the certificate name to **sssaaaa**.

- Example request

```
{
"modify_key":"CERT_NAME",
"modify_value": "sssaaaa"
}
```

- Example response

```
{
"response_info":"success"
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-7 lists the normal status code returned by the API.

Table 6-7 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.5 Querying the Product Type of a Certificate

Function

This API is used to query information about all products that are being sold on SCM.

URI

- URI format

  GET /v2/{project_id}/scm/cert/product

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
type_list	Yes	Array of type_list objects	Product type list. For details, see **Table 6-8**.

Table 6-8 type_list

Parameter	Mandatory	Type	Description
cert_type	Yes	String	Certificate type. • **OV_SSL_CERT**: Organization Validation (OV) SSL certificate. • **EV_SSL_CERT**: Extended Validation (EV) SSL certificate.
cert_brand	Yes	String	Certificate brand. **GLOBALSIGN**: GlobalSign brand.
domain_type	Yes	String	Domain name type. • **SINGLE_DOMAIN**: single-domain name type. • **MULTI_DOMAIN**: multi-domain name type. • **WILDCARD**: wildcard domain name type.
product_id	Yes	String	Product ID.

Parameter	Mandatory	Type	Description
effective_time	Yes	Integer	Certificate validity period (year). • **1**: The validity period of the certificate is one year. • **2**: The validity period of the certificate is two years.
product_name	Yes	String	Product name.

Example

- Example request

  None

- Example response

```
{
"type_list": [{
"cert_type": "OV_SSL_CERT",
"cert_brand":"GLOBALSIGN",
"domain_type":"SINGLE_DOMAIN",
"product_id":"00301-106005-0--0",
"effective_time":1,
"product_name ":"globalsign.single.ov.2"
}]
}
```

  or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
 }
```

Status Codes

Table 6-9 lists the normal status code returned by the API.

Table 6-9 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.6 Querying the Product Details of a Certificate

Function

This API is used to query details about a specified certificate.

URI

- URI format

  GET /v2/{project_id}/scm/product/{product_id}

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
product_id	Yes	String	Product ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
cert_type	Yes	String	Certificate type. • **OV_SSL_CERT**: Organization Validation (OV) SSL certificate. • **EV_SSL_CERT**: Extended Validation (EV) SSL certificate.
cert_brand	Yes	String	Certificate brand. **GLOBALSIGN**: GlobalSign brand.

Parameter	Mandatory	Type	Description
domain_type	Yes	String	Domain name type. • **SINGLE_DOMAIN**: single-domain name type. • **MULTI_DOMAIN**: multi-domain name type. • **WILDCARD**: wildcard domain name type.
effective_time	Yes	Integer	Certificate validity period, in years. • **1**: The validity period of the certificate is one year. • **2**: The validity period of the certificate is two years.

Example

- Example request

  None

- Example response

```
{
"cert_type": "OV_SSL_CERT",
"cert_brand":"GLOBALSIGN",
"domain_type":"SINGLE_DOMAIN",
"effective_time":1
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-10 lists the normal status code returned by the API.

Table 6-10 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.7 Applying for a Certificate

Function

This API is used to complete certificate application information, such as the domain name bound to a certificate and the applicant's detailed information.

NOTE

The request parameter **agree_privacy_protection** must be set to **true**. Otherwise, the certificate application information cannot be submitted.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/complete

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
domain	Yes	String	Domain name bound to a certificate. • If the certificate to be purchased is a single-domain or wildcard domain name certificate, enter the single-domain or wildcard domain name. • If the certificate to be purchased is a multi-domain certificate, select one domain name as the primary domain name. Example: www.example.com

Parameter	Mandatory	Type	Description
sans	No	String	Additional domain name of the certificate that is bound to a multi-domain certificate. Set this parameter only when the certificate to be purchased is a multi-domain certificate and the number of additional domain names can be increased. Multiple domain names must be separated by semicolons (;). Example: www.example.com;www.example1.com;www.example2.com
csr	No	String	Certificate CSR, which must match the domain name.
company_name	Yes	String	Company name. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
company_unit	No	String	Department name. This parameter is optional for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
company_province	Yes	String	State or region where a company is located. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.

Parameter	Mandatory	Type	Description
company_city	Yes	String	City where a company is located. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
country	Yes	String	Country code. • **CN**: China • **HK**: Hong Kong SAR, China • **US**: United States
applicant_name	Yes	String	Applicant name. The value is a string of 0 to 63 characters.
applicant_phone	Yes	String	Phone number of an applicant. Example: 13212345678
applicant_email	Yes	String	Email of an applicant. Example: example.huawei.com
contact_name	No	String	Name of a technical contact. The value is a string of 0 to 63 characters.
contact_phone	No	String	Phone number of a technical contact. Example: 13212345678
contact_email	No	String	Email of a technical contact. Example: example.huawei.com

Parameter	Mandatory	Type	Description
auto_dns_auth	No	Boolean	Whether to push DNS authentication information to HUAWEI CLOUD DNS. • **true**: DNS authentication information is pushed to HUAWEI CLOUD DNS. • **false**: DNS authentication information is not pushed to HUAWEI CLOUD DNS.
agree_privacy_protection	Yes	Boolean	Whether to agree with the privacy statement. • **true**: Agree with the privacy statement. • **false**: Disagree with the privacy statement. You can submit your certificate application only when this parameter is set to **true**.

Response

Response parameters

Parameter	Mandatory	Type	Description
request_info	Yes	String	Request result.

Example

The following describes how to supplement information about a certificate.

- Example request

```
{
   "domain":"www.xzz.com",
    "company_name": "Huawei Chengdu branch",
   "company_province": "Sichuan",
  "company_city": "Chengdu",
   "applicant_name": "Tom",
   "applicant_phone":"13212345678",
   "applicant_email":"9997342346@qq.com",
   "csr":"",
   "sans":"",
   "country":"CN",
   "company_unit": "Human Resource Dept",
```



```
    "contact_name": "Jacky",
    "contact_phone":"13512345678",
    "contact_email":"jk@jk.ff",
    "auto_dns_auth":false,
    "agree_privacy_protection":true
}
```

- Example response

```
{
    "request info":"success"
}
```

or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
 }
```

Status Codes

Table 6-11 lists the normal status code returned by the API.

Table 6-11 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.8 Verifying a CSR

Function

This API is used to verify a certificate signing request (CSR) and resolve the domain name.

URI

- URI format

  POST /v2/{project_id}/scm/check-csr

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
csr	Yes	String	Certificate signing request.

Response

Response parameters

Parameter	Mandatory	Type	Description
domain_name	Yes	String	Domain name in the CSR.

Example

The following describes how to verify a CSR.

- Example request

```
{
   "csr":"-----BEGIN NEW CERTIFICATE REQUEST-----******-----END NEW CERTIFICATE REQUEST-----"
}
```

- Example response

```
{
   "domain": "a.example1.com"
}
```

or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
 }
```

Status Codes

Table 6-12 lists the normal status code returned by the API.

Table 6-12 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.9 Saving Certificate Information

Function

This API is used to save certificate information entered during certificate application.

NOTE

The request parameter **agree_privacy_protection** must be set to **true**. Otherwise, certificate information cannot be saved.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/save

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
domain	Yes	String	Domain name bound to a certificate.
sans	No	String	Additional domain names of a multi-domain certificate. Multiple domain names are separated by semicolons (;).
csr	No	String	Certificate csr string, which must match the domain name.
company_name	Yes	String	Company name. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.

Parameter	Mandatory	Type	Description
company_unit	No	String	Department name. This parameter is optional for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
company_province	Yes	String	State or region where a company is located. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
company_city	Yes	String	City where a company is located. This parameter is mandatory for certificates of the OV and EV types. The value is a string of 0 to 63 characters.
country	Yes	String	Country code.
applicant_name	Yes	String	Applicant name. The value is a string of 0 to 63 characters.
applicant_phone	Yes	String	Phone number of an applicant. Example: 13212345678
applicant_email	Yes	String	Email of an applicant. Example: example.huawei.com
contact_name	No	String	Name of a technical contact. The value is a string of 0 to 63 characters.
contact_phone	No	String	Phone number of a technical contact. Example: 13212345678
contact_email	No	String	Email of a technical contact. Example: example.huawei.com

Parameter	Mandatory	Type	Description
agree_privacy_protection	Yes	Boolean	Whether to agree with the privacy statement. • **true**: Agree with the privacy statement. • **false**: Disagree with the privacy statement. You can save certificate information only when this parameter is set to **true**.

Response

Response parameters

Parameter	Mandatory	Type	Description
request_info	Yes	String	Request result.

Example

The following describes how to save supplemented information about a certificate.

- Example request

```
{
   "domain":"www.xzz.com",
    "company_name": "Huawei Chengdu branch",
   "company_province": "Sichuan",
  "company_city": "Chengdu",
   "applicant_name": "Tom",
   "applicant_phone":"13212345678",
   "applicant_email":"9997342346@qq.com",
   "csr":"",
   "sans":"",
   "country":"CN",
   "company_unit": "Human Resource Dept",
   "contact_name": "Jacky",
   "contact_phone":"13512345678",
   "contact_email":"jk@jk.ff",
   "agree_privacy_protection":true
}
```

- Example response

```
{
   "request info":"success"
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
 }
```


Status Codes

Table 6-13 lists the normal status code returned by the API.

Table 6-13 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.10 Reading the Information Entered When Applying for a Certificate

Function

This API is used to read the saved information about a certificate.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/read
- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
domain_name	Yes	String	Domain name bound to a certificate. Example: www.domain.com

Parameter	Mandatory	Type	Description
sans	Yes	String	Additional domain names of a multi-domain certificate. Multiple domain names are separated by semicolons (;). If a single-domain or wildcard domain certificate is applied for, the value of this parameter is empty.
CSR	Yes	String	Certificate signing request.
country	Yes	String	Country code. Example: • **CN**: China • **HK**: Hong Kong SAR, China • **US**: United States
company_name	Yes	String	Company name.
company_unit	Yes	String	Department name
company_province	Yes	String	State or region where a company is located. Example: Sichuan
company_city	Yes	String	City where a company is located. Example: Chengdu
applicant_name	Yes	String	Applicant name. Example: Tom
applicant_phone	Yes	String	Phone number of an applicant. Example: 13412345678
applicant_email	Yes	String	Email of an applicant. Example: example.huawei.com
contact_name	Yes	String	Name of a technical contact.
contact_phone	Yes	String	Phone number of a technical contact.

Parameter	Mandatory	Type	Description
contact_email	Yes	String	Email of a technical contact.
bl	Yes	String	Whether the picture of bank account opening permit has been uploaded. ● **0**: The picture of bank account opening permit has not been uploaded. ● **1**: The picture of bank account opening permit has been uploaded.
tl	Yes	String	Whether the business license of the company has been uploaded. **0**: The business license of the company has not been uploaded. **1**: The business license of the company has been uploaded.

Example

- Example request

  None

- Example response

```
{
   "domain_name": "www.xzz.com",
   "sans": "",
   "CSR": null,
   "country": "CN",
   "company_unit": "Human Resource Dept",
    "company_name": "Huawei Chengdu branch",
   "company_province": "Sichuan",
  "company_city": "Chengdu",
   "applicant_name": "Tom",
   "applicant_phone": "13245678932",
   "applicant_email": "1027342346@qq.com",
   "contact_name": "Jacky",
   "contact_phone": "13526456325",
   "contact_email": "jk@jk.ff",
   "bl": "0",
   "tl": "1"
}
```

  or

```
{
  "error_code": "SCM.XXXX",
```



```
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-14 lists the normal status code returned by the API.

Table 6-14 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.11 Canceling an Application

Function

This API is used to cancel an application of certificate reviewing.

URI

- URI format
  POST /v2/{project_id}/scm/cert/{cert_id}/cancel-cert
- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.
message	Yes	String	Request result.

Example

- Example request

  None

- Example response

```
{
   "cert_id": " scs1481110651012",
   "message": "success"
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
 }
```

Status Codes

Table 6-15 lists the normal status code returned by the API.

Table 6-15 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.12 Deleting a Certificate

Function

This API is used to delete a certificate, that is, delete a certificate from HUAWEI CLOUD.

NOTICE

This API will be discarded. You are advised to delete a certificate by referring to **Deleting a Certificate**.

URI

- URI format

  DELETE /v2/{project_id}/scm/cert/{cert_id}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Request result.

Example

- Example request

  None

- Example response

```
{
    "message": "success"
}
```

  or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
 }
```

Status Codes

Table 6-16 lists the normal status code returned by the API.

Table 6-16 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.13 Uploading Authentication Information

Function

This API is used to upload the authentication information picture required for certificate review.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/info/{type}/upload_authentication

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
type	Yes	String	Type of the content to be uploaded. • **BL**: bank account opening permit. • **TL**: business license of a company.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
request_info	Yes	String	Request result.

Example

- Example request

```
{
    <Upload content>
}
```

- Example response

```
{
   "request_info":"success"
}
```


or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-17 lists the normal status code returned by the API.

Table 6-17 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.14 Downloading a Certificate

Function

This API is used to download a certificate.

NOTICE

This API will be discarded. You are advised to export a certificate by referring to **Exporting a Certificate**.

URI

- URI format

  GET /v2/{project_id}/scm/cert/{cert_id}/cert_file
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Requests

Request parameters

None

Responses

Certificate file, which is a compressed package with the .rar extension.

Examples

- Example request

  None

- Example response

```
{
   <Object Content>
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-18 lists the normal status code returned by the API.

Table 6-18 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.15 Uploading a Certificate

Function

This API is used to upload a certificate to SCM.

URI

- URI format

  POST /v2/{project_id}/scm/cert/upload

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
cert_name	Yes	String	Certificate name. The value is a string of 0 to 63 characters.
cert	Yes	String	Certificate chain content.
private_key	Yes	String	Private key of a certificate.

Response

Response parameters

Parameter	Mandatory	Type	Description
cert_id	Yes	String	Certificate ID.

Example

The following describes how to upload a certificate named **test**.

- Example request

```
{
   "cert_name":"test",
   "cert":"-----BEGIN CERTIFICATE----- *** -----END CERTIFICATE-----",
   "private_key": "-----BEGIN RSA PRIVATE KEY----- *** -----END RSA PRIVATEKEY-----"
}
```

- Example response

```
{
   "cert_id": " scs1481110651012"
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
 }
```

Status Codes

Table 6-19 lists the normal status code returned by the API.

Table 6-19 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.16 Revoking a Certificate

Function

This API is used to revoke a certificate.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/revoke

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
reason	Yes	String	Reason for revoking a certificate. The value is a string of 0 to 63 characters.

Response

Response parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Revocation request result.

Examples

The following uses the certificate revocation reason "certificate information filled incorrectly" as an example.

- Example request

```
{
   "reason": "certificate information filled incorrectly",
}
```

- Example response


```
{
   "message":"success"
}
```

or

```
{
   "error_code": "SCM.XXXX",
   "error_msg": "XXXX"
}
```

Status Codes

Table 6-20 lists the normal status code returned by the API.

Table 6-20 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.17 Pushing a Certificate

Function

This API is used to push an SSL certificate to other HUAWEI CLOUD services, such as Web Application Firewall (WAF), Elastic Load Balance (ELB), and Content Delivery Network (CDN).

NOTICE

This API will be discarded. You are advised to push a certificate by referring to **Pushing a Certificate**.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/push

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
service_type	Yes	String	Type of the service to which a certificate is pushed. Options: **CDN**, **ELB**, **Enhance_ELB**, and **WAF**
remote_project	Yes	String	Region where the target service to which a certificate is pushed.

Response

Response parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Request result.

Example

The following describes how to push a certificate to WAF in region **cn-north-7**.

- Example request

```
{
   "service_type":"WAF",
   "remote_project":"cn-north-7"
}
```

- Example response

```
{
   "message":"success"
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
 }
```

Status Codes

Table 6-21 lists the normal status code returned by the API.

Table 6-21 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.18 Querying Push Records

Function

This API is used to query the last 10 certificate push records, which are to be pushed to another HUAWEI CLOUD service.

URI

- URI format

  GET /v2/{project_id}/scm/cert/{cert_id}/push-history

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
push_history_list	Yes	Array of push_history_list objects	Push record list. For details, see **Table 6-22**.

Table 6-22 push_history_list

Parameter	Mandatory	Type	Description
push_time	Yes	String	Push time, in milliseconds.
push_remote_project	Yes	String	Push project.

Parameter	Mandatory	Type	Description
push_service	Yes	String	Push service type. • **WAF**: A certificate is pushed to WAF. • **CDN**: A certificate is pushed to CDN. • **ELB**: A certificate is pushed to classic ELB. • **Enhance_ELB**: A certificate is pushed to an ELB load balancer (dedicated or shared load balancer).

Example

- Example request

  None

- Example response

```
{
   "push_history_list": [
      {
         "push_time": "1556257820000",
         "push_remote_project": null,
         "push_service": "CDN"
      },
      {
         "push_time": "1556257447000",
         "push_remote_project": "cn-north-7_test",
         "push_service": "WAF"
      }
   ]
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```

Status Codes

Table 6-23 lists the normal status code returned by the API.

Table 6-23 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.19 Canceling Authorization for Privacy Information

Function

This API is used to cancel authorization for privacy information and delete the privacy data saved in SCM.

URI

- URI format

  DELETE /v2/{project_id}/scm/privacy-protection/{cert_id}

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

None

Response

Response parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Request result.

Example

- Example request

  None

- Example response

```
{
   "message":"success"
}
```

  or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
}
```


Status Codes

Table 6-24 lists the normal status code returned by the API.

Table 6-24 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

6.20 Adding an Additional Domain Name

Function

This API is used to add an additional domain name. If you have a multi-domain SSL certificate and available quota for additional domain names, you can add additional domain names for the certificate after it is issued.

URI

- URI format

  POST /v2/{project_id}/scm/cert/{cert_id}/supplement

- Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
cert_id	Yes	String	Certificate ID.

Request

Request parameters

Parameter	Mandatory	Type	Description
ori_sans	Yes	String	Additional domain name bound to a multi-domain certificate. If multiple domain names are displayed, the domain names are separated by semicolons (;). Example: example.domain.com;example.domain1.com

Parameter	Mandatory	Type	Description
add_sans	No	String	Additional domain name to be added for a multi-domain certificate. If multiple domain names need to be entered, separate the domain names by semicolons (;). Example: example.domain2.com;example.domain3.com
email	No	String	Email of a contact.

Response

Response parameters

Parameter	Mandatory	Type	Description
request_info	Yes	String	Request result.

Example

The following describes how to add an additional domain name **example.domain.com**.

- Example request

```
{
   "ori_sans": "abc.com;xyz.com",
   "add_sans": "example.domain.com",
   "email": "example@xx.com"
}
```

- Example response

```
{
   "request info":"success"
}
```

or

```
{
  "error_code": "SCM.XXXX",
  "error_msg": "XXXX"
 }
```

Status Codes

Table 6-25 lists the normal status code returned by the API.

Table 6-25 Status code

Status Code	Status	Description
200	OK	Request processed successfully.

For details about error code, see **Error Code**

7 Permissions Policies and Supported Actions

7.1 Introduction to Permissions Policies and Supported Actions

This chapter describes fine-grained permissions management for your SCM. If your HUAWEI CLOUD account does not need individual IAM users, then you may skip over this chapter.

By default, new IAM users do not have permissions assigned. You need to add a user to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specified operations on cloud services based on the permissions.

Permissions are classified into **roles** and **policies** based on the authorization granularity. Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

Policy-based authorization is useful if you want to allow or deny the access to an API.

An account has all of the permissions required to call all APIs, but IAM users must have the required permissions specifically assigned. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user queries ECSs using an API, the user must have been granted permissions that allow the **ecs:servers:list** action.

Supported Actions

SCM provides system-defined policies that can be directly used in IAM. You can also create custom policies and use them to supplement system-defined policies, implementing more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permission: A statement in a policy that allows or denies certain operations.
- APIs: REST APIs that can be called in a custom policy
- Actions: Added to a custom policy to control permissions for specific operations.
- Dependent actions: When assigning an action to users, you also need to assign dependent permissions for that action to take effect.
- IAM projects or enterprise projects: Scope of users a permission is granted to. Policies that contain actions supporting both IAM and enterprise projects can be assigned to user groups and take effect in both IAM and Enterprise Management. Policies that only contain actions supporting IAM projects can be assigned to user groups and only take effect for IAM. Such policies will not take effect if they are assigned to user groups in Enterprise Management. For details about the differences between IAM and enterprise projects, see **What Are the Differences Between IAM and Enterprise Management?**

  NOTE

  √: supported; x: not supported

SCM supports the actions (shown in **API Actions**) that can be defined in custom policies. The actions include uploading, applying for, and downloading a certificate.

7.2 API Actions

Authorization information of APIs (v3)

Permission	API	Action	Dependent Permission	IAM Project (Project)	Enterprise Project (Enterprise Project)
Querying the certificate list	GET /v3/scm/certificates	scm:cert:list	-	√	x
Obtaining details of a certificate	GET /v3/scm/certificates/{certificate_id}	scm:cert:get	-	√	x
Deleting a certificate	DELETE /v3/scm/certificates/{certificate_id}	scm:cert:delete	-	√	x

Permission	API	Action	Dependent Permission	IAM Project (Project)	Enterprise Project (Enterprise Project)
Pushing a certificate	POST /v3/scm/certificates/{certificate_id}/push	scm:cert:push	The following action needs to be added when a certificate is to be pushed to CDN: cdn:configuration:queryHttpsConf	√	x
Importing a certificate	POST /v3/scm/certificates/import	scm:cert:upload	-	√	x
Exporting a certificate	POST /v3/scm/certificates/{certificate_id}/export	scm:cert:download	-	√	x

Authorization information of APIs (v2)

Permission	API	Action	Dependent Permission	IAM Project (Project)	Enterprise Project (Enterprise Project)
Querying the certificate list	GET /v2/{project_id}/scm/certlist	scm:cert:list	-	√	x
Querying details of a certificate	GET /v2/{project_id}/scm/cert/{cert_id}	scm:cert:get	-	√	x
Querying the certificate type	GET /v2/{project_id}/scm/cert/product	scm:certType:get	-	√	x
Querying details of a certificate	GET /v2/{project_id}/scm/product/{product_id}	scm:certProduct:get	-	√	x
Canceling an application	POST /v2/{project_id}/scm/cert/{cert_id}/cancel-cert	scm:cert:cancel	-	√	x
Purchasing a certificate	POST /v2/{project_id}/scm/cert/purchase	scm:cert:purchase	-	√	x
Applying for a certificate	POST /v2/{project_id}/scm/cert/{cert_id}/complete	scm:cert:complete	-	√	x
Saving the information entered when applying for a certificate	POST /v2/{project_id}/scm/cert/{cert_id}/save	scm:cert:complete	-	√	x

Permission	API	Action	Dependent Permission	IAM Project (Project)	Enterprise Project (Enterprise Project)
Reading the information entered during certificate application	POST /v2/{project_id}/scm/cert/{cert_id}/read	scm:cert:complete	-	√	x
Modifying a certificate	PUT /v2/{project_id}/scm/cert/{cert_id}	scm:cert:edit	-	√	x
Deleting a certificate	DELETE /v2/{project_id}/scm/cert/{cert_id}	scm:cert:delete	-	√	x
Downloading a certificate	GET /v2/{project_id}/scm/cert/{cert_id}/cert_file	scm:cert:download	-	√	x
Uploading authentication information	POST /v2/{project_id}/scm/cert/{cert_id}/info/{type}/upload_authentication	scm:cert:complete	-	√	x
Revoking a certificate	POST /v2/{project_id}/scm/cert/{cert_id}/revoke	scm:cert:revoke	-	√	x

Permission	API	Action	Dependent Permission	IAM Project (Project)	Enterprise Project (Enterprise Project)
Pushing a certificate	POST /v2/{project_id}/scm/cert/{cert_id}/push	scm:cert:push	The following action needs to be added when a certificate is to be pushed to CDN: cdn:configuration:queryHttpsConf	√	x
Querying push records	GET /v2/{project_id}/scm/cert/{cert_id}/push-history	scm:pushHistory:list	-	√	x
Uploading a certificate	POST /v2/{project_id}/scm/cert/upload	scm:cert:upload	-	√	x
Verifying a CSR	POST /v2/{project_id}/scm/check-csr	scm:cert:complete	-	√	x
Adding an additional domain name	POST /v2/{project_id}/scm/cert/{cert_id}/supplement	scm:cert:supplement	-	√	x
Canceling privacy authorization	DELETE /v2/{project_id}/scm/privacy-protection/{cert_id}	scm:privacyProtection:delete	-	√	x

A Appendix

A.1 Status Codes

Status Code	Status	Description
200	OK	Request processed successfully.
202	Accept	The job was successfully delivered. However, it will be postponed because the system is busy currently.
204	No Content	The request is processed successfully and no content is returned.
300	multiple choices	The requested resource has multiple available responses.
400	Bad Request	The request parameter is incorrect.
401	Unauthorized	You need to enter the username and password to access the requested page.
403	Forbidden	The server understood the request, but is refusing to fulfill it.
404	Not Found	The requested resource does not exist or not found.
405	Method Not Allowed	The method specified in the request is not allowed.
406	Not Acceptable	The response generated by the server cannot be accepted by the client.
407	Proxy Authentication Required	You must use the proxy server for authentication. Then, the request can be processed.

Status Code	Status	Description
408	Request Timeout	The request timed out.
409	Conflict	The request cannot be processed due to a conflict.
500	Internal Server Error	Internal service error.
501	Not Implemented	Failed to complete the request. The server does not support the requested function.
502	Bad Gateway	Failed to complete the request, because the server receives an invalid request.
503	Service Unavailable	Failed to complete the request due to system exception.
504	Gateway Timeout	A gateway timeout error occurs.

A.2 Error Codes

For more service error codes, see **API Error Center**.

Status Code	Error Codes	Error Message	Description	Solution
400	SCM.0005	Incorrect request parameter.	Incorrect request parameter	Enter a valid request parameter.
400	SCM.0008	Abnormal certificate ID	Abnormal certificate ID	Enter a correct certificate ID.
400	SCM.0009	Failed to upload the certificate because no domain name is bound to the certificate.	Failed to upload the certificate. No domain name is bound to the certificate.	Associate the domain name with the certificate before uploading.
400	SCM.0010	This operation is not allowed by the current certificate type or status.	The type or the current status of the certificate does not support this operation.	Enter a certificate type or a certificate status that supports this operation.

Status Code	Error Codes	Error Message	Description	Solution
400	SCM.0012	The uploaded private key failed to be resolved. Ensure that the certificate has been issued.	The uploaded private key failed to be resolved. Ensure that the uploaded certificate has been issued.	Enter a valid certificate private key.
400	SCM.0013	The uploaded certificate chain failed to be resolved. Ensure that the certificate has been issued.	The uploaded certificate chain failed to be resolved. Ensure that the uploaded certificate has been issued.	Enter a correct certificate chain.
400	SCM.0014	The uploaded certificate does not match the private key.	The uploaded certificate does not match the uploaded private key.	Upload a valid certificate and private key.
400	SCM.0015	The number or format of domain names entered does not meet the requirements of the your certificate.	The number or format of the domains you entered is not in accordance with requirements of your certificate.	Upload a single-domain or multiple-domain certificate.
400	SCM.0020	Incorrect certificate ID.	Incorrect cert ID.	Upload a valid certificate ID.
400	SCM.0030	Certificates cannot be pushed to this service.	Unsupported push service error	Enter a correct service name to which the certificate is pushed.
400	SCM.0031	Certificate parsing error.	Parse certificate error.	Upload a correct certificate.
400	SCM.0032	Incorrect certificate name.	Invalid certificate name.	Enter a valid certificate name.

Status Code	Error Codes	Error Message	Description	Solution
400	SCM.0059	The certificate private key is empty.	No certificate private key found.	Contact the customer service.
400	SCM.0069	No projects found.	No projects found.	Select a valid project.
400	SCM.0070	Incorrect format of the domain bound to the certificate to be uploaded.	The format of the domain bound to the certificate is incorrect.	Upload a valid certificate.
400	SCM.0201	Failed to push to ELB.	Failed to push to ELB. Eerror message returned.	Contact the customer service.
400	SCM.0202	Insufficient ELB permissions	Insufficient ELB permissions	Configure required ELB permissions.
400	SCM.0203	Failed to push to ELB.	Failed to push to ELB.	Contact the customer service.
400	SCM.0211	Failed to push to CDN.	Failed to push to CDN. Error message returned.	Contact the customer service.
400	SCM.0212	Insufficient CDN permissions	Insufficient CDN permissions	Configure required CDN permissions.
400	SCM.0221	Failed to push to WAF.	Failed to push to WAF. Eerror message returned.	Contact the customer service.
400	SCM.0222	Insufficient WAF permissions	WAF permission denied	Configure required WAF permissions.
401	SCM.1000	The token of the request fails to be authenticated	The token of the request fails to be authenticated	Enter a correct token.
401	SCM.1004	Failed to apply for OBT.	Failed to apply for OBT.	Apply for OBT.

Status Code	Error Codes	Error Message	Description	Solution
401	SCM.1009	Your account is restricted.	The account is restricted	Contact the administrator for unfreezing.
401	SCM.1010	Your account is frozen.	The account is frozen	Contact the administrator for unfreezing.
401	SCM.4002	You do not have permission to perform this operation.	You do not have permission to perform this operation.	Contact the administrator to obtain the permission.
403	SCM.0002	Incorrect tenant ID.	Incorrect tenant ID or domain ID	Ensure that the projectId that matches the projectId in X-Auth-Token.
500	SCM.0007	Failed to download the certificate.	Failed to download the certificate.	Contact the customer service.
500	SCM.0037	Failed to encrypt the certificate.	Failed to encrypt the certificate.	Contact the customer service.
500	SCM.0038	Failed to decrypt the certificate.	Failed to decrypt the certificate.	Contact the customer service.
500	SCM.0213	Failed to push to CDN.	CDN push failed	Contact the customer service.
500	SCM.0223	Failed to push to WAF.	WAF push failed	Contact the customer service.
500	SCM.4001	An error occurred when accessing the PDP API.	Failed to obtain fine-grained permission.	Contact the 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 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 **Regions and Endpoints**. For details about API authentication, see **Authentication**.

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

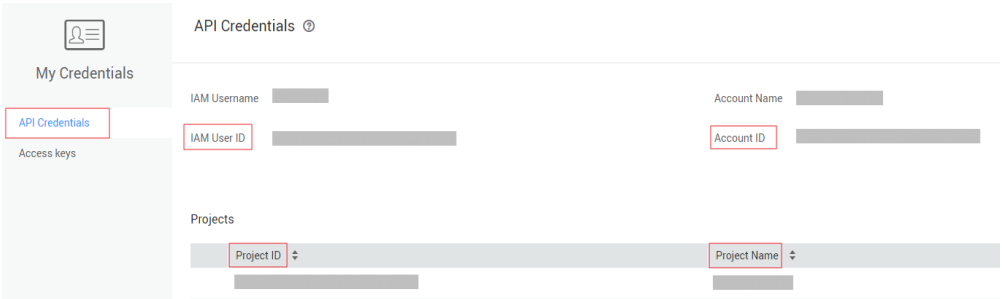
```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "xxxxxxxx",
            "description": "",
            "links": {
                "next": null,
                "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 **Basic Information** from the drop-down list.
3. On the **Account Info** page, click **Manage** next to **Security Credentials**.
   On the **API Credentials** page, view project IDs in the project list.

Figure A-1 Viewing project IDs



B Change History

Released On	Description
2021-05-07	This issue is the tenth official release. Optimized the parameter description in "Error Codes."
2021-02-20	This issue is the ninth official release. Modified the parameter description in "Importing a Certificate."
2020-12-10	This issue is the eighth official release. Added v3 API authorization information in **API Actions**.
2020-11-24	This issue is the seventh official release. Added **Examples**.
2020-10-09	This issue is the sixth official release. Released the new APIs.
2020-08-10	This issue is the fifth official release. • Changed the title of section from "Restrictions" to **Constraints**. • Modified **Making an API Request**. • Added the description about whether the service requires a global or project-level token in section **Authentication**.
2020-06-30	This issue is the fourth official release. Optimized **Permissions Policies and Supported Actions**.

Released On	Description
2020-01-20	This issue is the third official release. Updated descriptions in section "Permissions and Supported Actions" based on the changes on the IAM console.
2019-09-11	This is the second official release. Optimized section "Obtaining a Project ID."
2019-08-13	This issue is the first official release.

