Simple Message Notification

API Reference

Issue 01
Date 2025-01-07

HUAWEI TECHNOLOGIES CO., LTD.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Before You Start

Welcome to the *Simple Message Notification API Reference*. Simple Message Notification (SMN) is a reliable and flexible large-scale message notification service. It enables you to efficiently send messages to email addresses, phone numbers, and HTTP/HTTPS servers and connect cloud services through notifications, reducing system complexity.

You can use APIs described in this document to manage SMN resources, such as creating, querying, deleting, and updating resources. For details about all supported operations, see **API Overview**.

If you plan to access SMN through an API, ensure that you are familiar with basic SMN concepts. For details, see "Service Overview" in the *Simple Message Notification User Guide*.

Endpoints

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

Concepts

- Account

  An account is created upon successful signing up. 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, which should not be used to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- User

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

  API authentication requires information such as the account name, username, and password.

- Region

  A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other

over an intranet, while AZs in different regions are isolated from each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

- AZ

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

- Project

  A project corresponds to a region. Default projects are defined to group and physically isolate resources (including computing, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

  Figure 1-1 Project isolating model
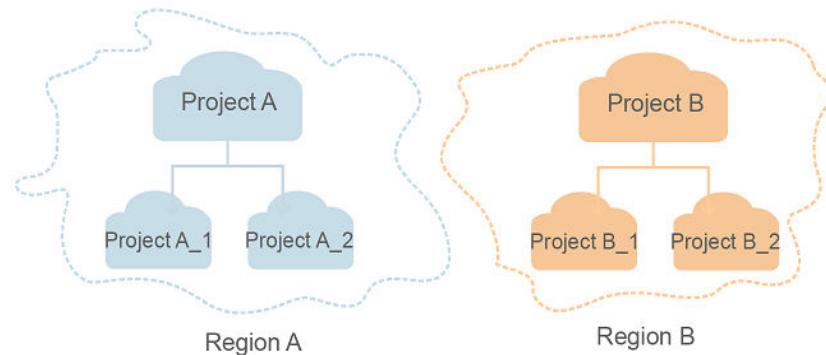
2 API Overview

By calling SMN APIs, you can use all functions provided by SMN, for example, creating a topic and adding a subscription.

Table 2-1 API description

Category	Description
Topic operations	Create, update, and delete a topic.
Subscription operations	Query the subscription list, add subscriptions, and delete subscriptions.
Template operations	Create, modify, and delete message templates.
Message publishing	Publish a message, publish a probing message, and obtain the HTTP probing result.
Resource tag operations	Query resources by tag, add resource tags, and query tags.
Version querying	Query SMN API versions and information about the SMN API v2 version.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

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

Table 3-1 URI parameter description

Parameter	Description
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**.
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of *Parameter name=Parameter value*. For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

📖 **NOTE**

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

Request Methods

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

Table 3-2 HTTP methods

Method	Description
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://{{endpoint}}/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.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of *Hostname:Port number*. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value **application/json** is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in **Obtaining a Project ID**.	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4

Parameter	Description	Mandatory	Example Value
X-Auth-Token	Specifies the user token. It is a response to the API for obtaining a user token (This is the only API that does not require authentication). After the request is processed, the value of **X-Subject-Token** in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvc NAQcCo...ggg1B BIINPXsidG9rZ

☐ **NOTE**

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added in the request.

For more details, see "Authentication Using AK/SK" in **Authentication**.

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

```
POST https://{{endpoint}}/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format (for example, JSON or XML) 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. Replace *username*, *domainname*, *$ADMIN_PASS* (login password), and *xxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from **Regions and Endpoints**.

☐ **NOTE**

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. 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://{{endpoint}}/v3/auth/tokens
Content-Type: application/json
```

```
{
   "auth": {
      "identity": {
         "methods": [
            "password"
         ],
         "password": {
            "user": {
               "name": "username",
               "password": "$ADMIN_PASS",     //You are advised to store it in ciphertext in the
configuration file or an environment variable and decrypt it when needed to ensure security.
               "domain": {
                  "name": "domainname"
               }
            }
         }
      },
      "scope": {
         "project": {
            "name": "xxxxxxxxxxxxxxxxxx"
         }
      }
   }
}
```

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

3.2 Authentication

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

- AK/SK authentication: Requests are encrypted using AK/SK pairs. AK/SK authentication is recommended because it is more secure than token authentication.

- Token authentication: Requests are authenticated using tokens.

AK/SK Authentication

☐☐ NOTE

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

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

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.

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

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or using the signing SDK. For details about how to sign requests and use the signing SDK, see **API Request Signing Guide**.

📖 **NOTE**

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

Token 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. You can obtain a token by calling the Obtaining User Token API.

IMS is a project-level service. When you call the API, set **auth.scope** in the request body to **project**.

```
{
   "auth": {
      "identity": {
         "methods": [
            "password"
         ],
         "password": {
            "user": {
               "name": "username",   // IAM user name
               "password": $ADMIN_PASS, //IAM user password. You are advised to store it in ciphertext in
the configuration file or an environment variable and decrypt it when needed to ensure security.
               "domain": {
                  "name": "domainname"  // Name of the account to which the IAM user belongs
               }
            }
         }
      },
      "scope": {
         "project": {
            "name": "xxxxxxxx"    // Project name
         }
      }
   }
}
```

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

```
POST https://{{endpoint}}/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

3.3 Response

Status Code

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

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see **Returned Value**.

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

Response Header

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

Figure 3-1 shows the response header fields for the API used to obtain a user token. The **X-Subject-Token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

> ☐ **NOTE**
>
> For security purposes, you are advised to set the token in ciphertext in configuration files or environment variables and decrypt it when using it.

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


```
connection →  keep-alive

content-type →  application/json

date →  Tue, 12 Feb 2019 06:52:13 GMT

server →  Web Server

strict-transport-security →  max-age=31536000; includeSubdomains;

transfer-encoding →  chunked

via →  proxy A

x-content-type-options →  nosniff

x-download-options →  noopen

x-frame-options →  SAMEORIGIN

x-iam-trace-id →  218d45ab-d674-4995-af3a-2d0255ba41b5

x-subject-token
→                                                                              ID
fj3K                                                                           EI
xHR
j+Cl
RzT0w0bpvGw-0PNFYxLCKhorShiR0zv0vN--nSd0NbXg--

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

(Optional) Response Body

The body of a response is often returned in a structured format (for example, JSON or XML) as specified in the **Content-Type** header field. The response body transfers content except the response header.

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

```
{
   "token": {
      "expires_at": "2019-02-13T06:52:13.855000Z",
      "methods": [
         "password"
      ],
      "catalog": [
         {
            "endpoints": [
               {
                  "region_id": "az-01",
......
```

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

```
{
    "error_msg": "The request message format is invalid.",
    "error_code": "IMG.0001"
}
```

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

4 Getting Started

Scenarios

This topic describes how to publish a message to a topic by calling APIs.

For details about how to call APIs, see **Calling APIs**.

> **NOTE**
>
> The token obtained from IAM is valid for only 24 hours. If you want to use one token for authentication, you can cache it to avoid frequently calling the IAM API.

Involved APIs

To use token authentication, you need to obtain a token and add **X-Auth-Token** to the request header of API calls.

- IAM API to obtain a token
- SMN API to create a topic
- SMN API to add subscriptions
- SMN API to publish messages

Procedure

1. Obtain the token by referring to **Authentication**.
2. Send a request **POST https://{SMN_Endpoint}/v2/{project_id}/ notifications/topics** to create a topic.

   a. Add **X-Auth-Token** to the request header.

   b. Specify the following parameters in the request body:
   ```
   {
      "name": "test_topic_v2", // Topic name (mandatory)
      "display_name": "test"  // Display name (optional)
   }
   ```
   If the request is successful, a request ID (**request_id**) and topic URN (**topic_urn**) are returned.

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

3. Send a request **POST https://{SMN_Endpoint}/v2/{project_id}/ notifications/topics/{topic_urn}/subscriptions** to add subscriptions.

Set **topic_urn** based on the response in **2.b**.

a.   Add **X-Auth-Token** to the request header.

b.   Specify the following parameters in the request body:
```
{
  "protocol": "email", // Subscription protocol (mandatory)
  "endpoint": "xxx@xx.com", // Email address (mandatory)
  "remark": "Subscription" // Description (optional)
}
```

If the request is successful, a request ID (**request_id**) and subscription URN (**subscription_urn**) are returned.

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

📖 NOTE

You can add multiple kinds of subscriptions to a topic. For details, see **Adding a Subscription**.

c.   The subscription endpoint receives a confirmation message and must confirm the subscription in the email.

4.   Send a request **POST https://{SMN_Endpoint}/v2/{project_id}/ notifications/topics/{topic_urn}/publish** to publish a message.

a.   Add **X-Auth-Token** to the request header.

b.   Specify the following parameters in the request body:
```
{
  "subject":"test message v2", // Message subject, which is displayed in email messages
(optional)
  "message": "Message test message v2." // Message to be published (mandatory)
}
```

If the request is successful, a request ID (**request_id**) and message ID (**message_id**) are returned.

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

📖 NOTE

Only confirmed subscription endpoints can receive the message you publish.

<div align="right">

5 APIs

</div>

5.1 Topic Operations

5.1.1 Creating a Topic

Description

- API name

  CreateTopic

- Function

  Create a topic. Each user can create 3,000 topics at most. In the high-concurrent scenario, a user may create a few topics more than 3,000.

  The API is idempotent. It returns a successful result after creating a topic. If a topic of the same name already exists, the status code is 200. Otherwise, the status code is 201.

URI

- URI format

  POST /v2/{project_id}/notifications/topics

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the topic Enter 1 to 255 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The topic name must start with a letter or digit.
display_name	Yes	String	Topic display name, which is presented as the name of the email sender in email messages The display name cannot exceed 192 bytes. **display_name** is left blank by default.
enterprise_project_id	No	String	(Optional) Enterprise project ID. It is required when the enterprise project function is enabled. Default value: **0**

- Example request

  POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics
  ```
  {
      "name": "test_topic_v2",
      "display_name": "testtest"
  }
  ```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
topic_urn	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

- Example response

  ```
  {
      "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
      "topic_urn": "urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2"
  }
  ```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.2 Updating a Topic

Description

- API name

  UpdateTopic
- Function

  Update the topic display name.

URI

- URI format

  PUT /v2/{project_id}/notifications/topics/{topic_urn}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
display_name	Yes	String	Topic display name, which is presented as the name of the email sender in email messages The display name cannot exceed 192 bytes.

- Example request

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2
{
    "display_name": "testtest222"
}
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response

```
{
    "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.3 Querying Topics

Description

- API name

  ListTopics

- Function

  Query the topics by page. The topics are sorted by the creation time in descending order. When querying the topics by page, you can specify **offset** and **limit**. If no topic has been created, an empty list is returned.

URI

- URI format

  GET /v2/{project_id}/notifications/topics?offset={offset}&limit={limit}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset If the offset is an integer greater than 0 but less than the number of resources, all resources after this offset will be queried. The default offset is **0**.
limit	No	Integer	• Number of resources returned on each page • Value range: 1–100 Commonly used numbers are **10**, **20**, and **50**. The default number is **100**.

Request

Example request
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics?offset=0&limit=100

Response

● Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
topic_count	Integer	Number of topics in your account **NOTE** No matter what you have set for **offset** and **limit** in the request, this API always returns the total number of topics in your account.
topics	Array of **Table 5-1** objects	Topic structure array

Table 5-1 Topic structure

Parameter	Type	Description
topic_urn	String	Resource identifier of a topic, which is unique
name	String	Name of the topic
display_name	String	Topic display name, which is presented as the name of the email sender in email messages
push_policy	Integer	Message push policy ● **0**: Failed messages will be saved in message queues. ● **1**: Failed messages will be discarded.
enterprise_project_id	String	Enterprise project ID

● Example response

```
{
    "request_id": "70bb40bef50e4a14b116a5a527fd7432",
    "topic_count": 1,
    "topics": [
        {
            "topic_urn": "urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:test_topic_v2",
            "display_name": "testtest",
            "name": "test_topic_v1",
            "push_policy": 0,
            "enterprise_project_id" : "0"
        }
    ]
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.4 Querying Details of a Topic

Description

● API name

QueryTopicDetail

● Function

Query the detailed information about a topic.

URI

- URI format

  GET /v2/{project_id}/notifications/topics/{topic_urn}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

Request

Example request

GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:test_create_topic_v2

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
name	String	Name of the topic
topic_urn	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.
display_name	String	Topic display name, which is presented as the name of the email sender in email messages
push_policy	Integer	Message pushing policy **0** indicates that the message sending fails and the message is cached in the queue. **1** indicates that the failed message is discarded.
create_time	String	Time when the topic was created The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.

Parameter	Type	Description
update_time	String	Time when the topic was updated The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.
enterprise_project _id	String	Enterprise project ID

- Example response

```
{
    "update_time": "2016-08-01T02:16:38Z",
    "push_policy": 0,
    "create_time": "2016-08-01T02:16:38Z",
    "name": "test_create_topic_v2",
    "topic_urn": "urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:test_create_topic_v2",
    "display_name": "test create topic v2",
    "request_id": "6837531fd3f54550927b930180a706bf",
    "enterprise_project_id" : "0"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.5 Querying a Topic Policy

Description

- API name

  ListTopicAttributes

- Function

  Query a topic policy.

URI

- URI format

  GET /v2/{project_id}/notifications/topics/{topic_urn}/attributes?name={name}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.
name	No	String	Specifies the policy name. Only specified policy names are supported. For details, see **Topic Attribute List**.

☐ NOTE

If **name** is not specified, all attribute values of the topic are queried. The supported attribute values are provided in **Topic Attribute List**.

Request

Example request
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:test_create_topic_v2/attributes?name=access_policy

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
attributes	Map	Attribute key-value pair **access_policy**: topic access policy **introduction**: description of a topic

- Example response

```
{
  "request_id": "6837531fd3f54550927b930180a706bf",
  "attributes": {
  "access_policy": "{
      "Version": "2016-09-07",
      "Id": "__default_policy_ID",
      "Statement": [
        {
          "Sid": "__user_pub_0",
          "Effect": "Allow",
          "Principal": {
            "CSP": [
```

```
                "urn:csp:iam::93dc1b4697ac493d9b7d089569f86b32:root"
              ]
            },
        "Action": ["SMN:Publish","SMN:QueryTopicDetail"],
        "Resource": "urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:aaa"
        },
        {
        "Sid": "__service_pub_0",
        "Effect": "Allow",
        "Principal": {
          "Service": ["obs"]
          },
        "Action": ["SMN:Publish","SMN:QueryTopicDetail"],
        "Resource": "urn:smn:regionId:8bad8a40e0f7462f8c1676e3f93a8183:aaa"
        }
      ]
    }"
  }
}
```

📖 **NOTE**

The value of **access_policy** is a JSON character string, which requires escape characters. While in the preceding example, the characters are not escaped. Escape them before using the policy.

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.6 Updating a Topic Policy

Description

- API name

  UpdateTopicAttribute

- Function

  Update a topic policy.

URI

- URI format

  PUT /v2/{project_id}/notifications/topics/{topic_urn}/attributes/{name}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.
name	Yes	String	Specifies the policy name. Only specified policy names are supported. For details, see **Topic Attribute List**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
value	Yes	String	Topic attribute value • If you set **name** to **access_policy**, **value** is the topic attributes, and the maximum length of **value** is 30 KB. • If you set **name** to **introduction**, **value** is the topic introduction, and the maximum length of **value** is 120 bytes. For details, see **Table 5-2**.

Table 5-2 Topic attribute values

Parameter	Description	Constraint
Version	Policy specification version	Only **2016-09-07** is supported.
Id	Policy ID, which uniquely identifies a policy	**Id** must be specified.

Parameter	Description	Constraint
Statement	Statements used to configure a topic policy. Each topic policy may contain one or more statements. You can use statements to grant topic permissions to other users or cloud services.	A policy must contain at least one statement. For details about elements in a statement, see **Table 5-3**.

Table 5-3 Statement elements description

Element	Description	Constraint
Sid	Statement ID	The statement ID must be unique, for example, **statement01** or **statement02**.
Effect	Statement effect	The effect can be **Allow** or **Deny**.
Principal NotPrincipal	• **Principal**: object to which the statement applies • **NotPrincipal**: object to which the statement does not apply The following two types of objects are supported: – **CSP**: one or more cloud users – **Service**: one or more cloud services	Either the **Principal** or **NotPrincipal** element must be configured. • If you enter **CSP**, you must specify user information in the format **urn:csp:iam::domainId:root**. Obtain the account ID of each user you specify. • If you enter **Service**, you must specify the cloud service names in lower case.

Element	Description	Constraint
Action NotAction	**Action**: allowed statement action **NotAction**: statement action not allowed You can use a wildcard character to configure a set of actions, for example, **SMN:Update*** and **SMN:Delete***. If you only enter a wildcard character (*) in a statement, all supported actions are allowed.	Either **Action** or **NotAction** must be configured. The following actions are supported: - SMN:UpdateTopic - SMN:DeleteTopic - SMN:QueryTopicDetail - SMN:ListTopicAttributes - SMN:UpdateTopicAttribute - SMN:DeleteTopicAttributes - SMN:DeleteTopicAttributeBy-Name - SMN:ListSubscriptionsByTopic - SMN:Subscribe - SMN:Unsubscribe - SMN:Publish
Resource	Specifies the topic the statement applies to.	Either **Resource** or **NotResource** must be configured. You need to enter a topic URN.
NotResource	Specifies the topic the statement does not apply to.	Either **Resource** or **NotResource** must be configured. You need to enter a topic URN.

- Example request

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/attributes/access_policy
{
  "value": "{
      \"Version\": \"2016-09-07\",
      \"Id\": \"__default_policy_ID\",
      \"Statement\": [
        {
         \"Sid\": \"__user_pub_0\",
         \"Effect\": \"Allow\",
         \"Principal\": {
          \"CSP\": [
                \"urn:csp:iam::{domainID}:root\"
                ]
          },
        \"Action\": [\"SMN:Publish\",\"SMN:QueryTopicDetail\"],
        \"Resource\": \"{topic_urn}\"
        },
        {
        \"Sid\": \"__service_pub_0\",
        \"Effect\": \"Allow\",
        \"Principal\": {
          \"Service\": [\"obs\"]
          },
        \"Action\": [\"SMN:Publish\",\"SMN:QueryTopicDetail\"],
        \"Resource\": \"{topic_urn}\"
        }
        ]
```

```
        }"
}
```

📖 **NOTE**

Replace **{project_id}**, **{domainID}**, and **{topic_urn}** with the actual values.

domainID indicates the user's account ID. To obtain it, log in to the SMN console, click **My Credentials** in the username drop-down list on the upper right.

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response
```
{
    "request_id":"6837531fd3f54550927b930180a706bf"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.7 Deleting a Topic

Description

- API name

  DeleteTopic

- Function

  Delete a topic and its subscribers. If a topic is deleted, a pending message will fail to deliver to the topic subscribers.

URI

- URI format

  DELETE /v2/{project_id}/notifications/topics/{topic_urn}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

Request

Example request
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response
```
{
    "request_id": "5fcba32bd2814ea39431829c22bda94b"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.8 Deleting a Specified Topic Policy

Description

- API name

  DeleteTopicAttributeByName

- Function

  Delete a specified topic policy.

URI

- URI format

  DELETE /v2/{project_id}/notifications/topics/{topic_urn}/attributes/{name}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.
name	Yes	String	Attribute name Only specified attribute names are supported. For details, see **Topic Attribute List**.

Request

Example request
```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/attributes/access_policy
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response
```
{
    "request_id": "6837531fd3f54550927b930180a706bf"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.1.9 Deleting All Topic Policies

Description

- API name

DeleteTopicAttributes

- Function

  Delete all topic policies.

URI

- URI format

  DELETE /v2/{project_id}/notifications/topics/{topic_urn}/attributes

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

Request

Example request
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/attributes

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response

```
{
    "request_id": "6837531fd3f54550927b930180a706bf"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.2 Subscription Operations

5.2.1 Adding a Subscription

Description

- API name

  Subscribe

- Function

  Add a subscription to a specified topic. If the status of the subscription is unconfirmed, a confirmation message is sent to the subscriber. After confirming the subscription, the subscriber can receive notification messages published to the topic.

  By default, 10,000 subscriptions can be added to a topic. However, in a high-concurrency scenario, which is rare, extra subscriptions may be added successfully.

  The API is idempotent. If the added subscription already exists, a successful result and status code 200 are returned. Otherwise, the status code is 201.

URI

- URI format

  POST /v2/{project_id}/notifications/topics/{topic_urn}/subscriptions

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
endpoint	Yes	String	Message endpoint **NOTE** For an HTTP subscription, the endpoint starts with **http://**. For an HTTPS subscription, the endpoint starts with **https://**. For an email subscription, the endpoint is an email address. For an SMS subscription, the endpoint is a phone number.
protocol	Yes	String	Subscription protocol (Different protocols indicate different types of endpoints to receive messages.) The following protocols are supported: ● **email**: The endpoints are email address. ● **sms**: The endpoints are phone numbers. ● **http** and **https**: The endpoints are URLs.
remark	No	String	Description of the subscription The remarks must be UTF-8-coded and can contain up to 128 bytes.

- Example request

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1/subscriptions
{
    "protocol": "email",
    "endpoint": "xxx@xxx.com",
    "remark": "O&M"
}
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

Parameter	Type	Description
subscription_urn	String	Resource identifier of a subscription, which is unique

- Example response

```
{
   "request_id": "fdbabe38ead6482b8574f82a3d1168e9",
   "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c3
77837"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.2.2 Querying Subscriptions

Description

- API name

  ListSubscriptions

- Function

  Query the list of all subscriptions by page. The list is sorted by the subscription adding time in ascending order. When querying the subscription list by page, you can specify **offset** and **limit**. If no subscription has been added to the topic, an empty list is returned.

URI

- URI format

  GET /v2/{project_id}/notifications/subscriptions?offset={offset}&limit={limit}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset If the offset is an integer greater than 0 but less than the number of resources, all resources after this offset will be queried. The default offset is **0**.
limit	No	Integer	● Value range: 1–100 Commonly used numbers are **10**, **20**, and **50**. ● Number of resources returned on each page The default number is **100**.

Request

Example request
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/subscriptions?offset=0&limit=2

Response

● Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
subscription_count	Integer	Number of subscriptions
subscriptions	Array of **Table 5-4** objects	Subscription structure

Table 5-4 Subscription structure

Parameter	Type	Description
topic_urn	String	Resource identifier of a topic, which is unique

Parameter	Type	Description
protocol	String	Subscription protocol (Different protocols indicate different types of endpoints to receive messages.) The following protocols are supported: ● **email**: The endpoints are email address. ● **sms**: The endpoints are phone numbers. ● **http** and **https**: The endpoints are URLs.
subscription_urn	String	Resource identifier of a subscription, which is unique
owner	String	Project ID of the topic creator
endpoint	String	Message receiving endpoint
remark	String	Remarks
status	Integer	Subscription status ● **0**: unconfirmed ● **1**: confirmed ● **3**: canceled

● Example response

```
{
    "request_id": "4650b14bf221492fb819c231d167e6fe",
    "subscription_count": 2,
    "subscriptions": [
        {
            "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
            "protocol": "sms",
            "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c3
77837",
            "owner": "762bdb3251034f268af0e395c53ea09b",
            "endpoint": "xxxxxxxxxxx",
            "remark": "",
            "status": 0
        },
        {
            "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
            "protocol": "email",
            "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:a2d52a9f5c3b47f48c3fafb177a5
8796",
            "owner": "762bdb3251034f268af0e395c53ea09b",
            "endpoint": "xx@xx.com",
            "remark": "",
            "status": 0
        }
```

```
        ]
    }
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.2.3 Querying Subscriptions of a Specified Topic

Description

- API name

  ListSubscriptionsByTopic

- Function

  Query the list of subscriptions of a specified topic by page. The list is sorted by the subscription adding time in ascending order. You can specify **offset** and **limit**. If no subscription has been added to the topic, an empty list is returned.

URI

- URI format

  GET /v2/{project_id}/notifications/topics/{topic_urn}/subscriptions?offset={offset}&limit={limit}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Topics**.
offset	No	Integer	Offset If the offset is an integer greater than 0 but less than the number of resources, all resources after this offset will be queried. The default offset is **0**.

Parameter	Mandatory	Type	Description
limit	No	Integer	- Number of resources returned on each page - Value range: 1–100 Commonly used numbers are **10**, **20**, and **50**. The default number is **100**.

Request

Example request
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1/subscriptions?offset=0&limit=100

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
subscription_count	Integer	Number of subscriptions
subscriptions	Array of **Table 5-5** objects	Subscription structure

Table 5-5 Subscription structure

Parameter	Type	Description
topic_urn	String	Resource identifier of a topic, which is unique

Parameter	Type	Description
protocol	String	Subscription protocol (Different protocols indicate different types of endpoints to receive messages.) The following protocols are supported: ● **email**: The endpoints are email address. ● **sms**: The endpoints are phone numbers. ● **http** and **https**: The endpoints are URLs.
subscription_urn	String	Resource identifier of a subscription, which is unique
owner	String	Project ID of the topic creator
endpoint	String	Message receiving endpoint
remark	String	Remarks
status	Integer	Subscription status ● **0**: unconfirmed ● **1**: confirmed ● **3**: canceled

● Example response

```
{
    "request_id": "4650b14bf221492fb819c231d167e6fe",
    "subscription_count": 2,
    "subscriptions": [
        {
            "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
            "protocol": "sms",
            "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c3
77837",
            "owner": "762bdb3251034f268af0e395c53ea09b",
            "endpoint": "xxxxxxxxxx",
            "remark": "",
            "status": 0
        },
        {
            "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
            "protocol": "email",
            "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:a2d52a9f5c3b47f48c3fafb177a5
8796",
            "owner": "762bdb3251034f268af0e395c53ea09b",
            "endpoint": "xx@xx.com",
            "remark": "",
            "status": 0
        }
```

```
    ]
  }
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.2.4 Deleting a Subscription

Description

- API name

  Unsubscribe

- Function

  Delete a specified subscription.

URI

- URI format

  DELETE /v2/{project_id}/notifications/subscriptions/{subscription_urn}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
subscription_urn	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Subscriptions**.

Request

Example request
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/subscriptions/
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response

```
{
    "request_id": "f3197b274a6b473a8007eed79e716c30"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.3 Template Operations

5.3.1 Creating a Message Template

Description

- API name

  CreateMessageTemplate

- Function

  Create a message template for quick message sending to reduce the request data volume.

  By default, a user can create a maximum of 100 message templates. However, in a high-concurrency scenario, which is rare, extra templates may be successfully created.

URI

- URI format

  POST /v2/{project_id}/notifications/message_template

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
message_template_name	Yes	String	Template name Enter 1 to 64 characters, and start with a letter or digit. Only letters, digits, hyphens (-), and underscores (_) are allowed.
content	Yes	String	Template content, which currently supports plain text only The template content cannot be left blank or larger than 256 KB.
protocol	Yes	String	Protocol supported by the template Currently, the following protocols are supported: ● **email** ● **default** ● **sms** ● **http** and **https**

- Example request

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template
{
    "message_template_name": "confirm_message",
    "protocol": "https",
    "content": "(1/2)You are invited to subscribe to topic({topic_id}). Click the following URL to confirm
subscription:(If you do not want to subscribe to this topic, ignore this message.)"
}
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
message_template_id	String	Resource identifier of the template, which is unique

- Example response

```
{
    "request_id": "ca03efa691624d8eb2dfeba01a1bcf6e",
    "message_template_id": "57ba8dcecda844878c5dd5815b65d10f"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.3.2 Modifying a Message Template

Description

- API name

  UpdateMessageTemplate
- Function

  Modify the message template content.

URI

- URI format

  PUT /v2/{project_id}/notifications/message_template/{message_template_id}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
message_templat e_id	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Message Templates**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
content	Yes	String	Template content, which currently supports plain text only
			The template content cannot be left blank or larger than 256 KB.

- Example request
```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template/
b3ffa2cdda574168826316f0628f774f
{
    "content": "(1/22)You are invited to subscribe to topic({topic_id_id1}). Click the following URL to
confirm subscription:(If you do not want to subscribe to this topic, ignore this message.)"
}
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response
```
{
    "request_id": "5fcba32bd2814ea39431829c22bda94b"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.3.3 Querying Message Templates

Description

- API name

  ListMessageTemplates

- Function

  Query the template list by page. The list is sorted by the template creation time in ascending order. You can specify **offset** and **limit**. If no template has been created, an empty list is returned. The parameters **message_template_name** and **protocol** are added.

📖 **NOTE**

The template list provides only brief information. To query the template details, see **Querying the Details of a Message Template**.

URI

- URI format

  GET /v2/{project_id}/notifications/message_template?
  offset={offset}&limit={limit}&message_template_name={message_template_name}&protocol={protocol}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
offset	No	Integer	Offset If the offset is an integer greater than 0 but less than the number of resources, all resources after this offset will be queried. The default offset is **0**.
limit	No	Integer	• Number of resources returned on each page • Value range: 1–100 Commonly used numbers are **10**, **20**, and **50**. The default number is **100**.
message_template_name	No	String	Template name Enter 1 to 64 characters, and start with a letter or digit. Only letters, digits, hyphens (-), and underscores (_) are allowed.

Parameter	Mandatory	Type	Description
protocol	No	String	Protocol supported by the template The following protocols are supported: ● **email** ● **default** ● **sms** ● **http** and **https**

Request

Example request
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template?
offset=0&limit=2&message_template_name=test1&protocol=email

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
message_template_count	String	Number of returned templates
message_templates	Array of **Table 5-6** objects	**Message_template** structure array

Table 5-6 Message template structure

Parameter	Type	Description
message_template_id	String	Template ID
message_template_name	String	Template name
protocol	String	Protocol supported by the template The following protocols are supported: ● **email** ● **default** ● **sms** ● **http** and **https**

Parameter	Type	Description
tag_names	String array	Variable list
create_time	String	Time when the template was created The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.
update_time	String	Last time when the template was updated The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.

- Example response

```
{
    "message_templates":[
        {
            "message_template_name": "confirm_message",
            "update_time": "2016-08-02T08:22:18Z",
            "create_time": "2016-08-02T08:22:18Z",
            "tag_names":[
                "topic_urn"
            ],
            "message_template_id": "79227dfdf88d4e52a1820ca1eb411635"
        },
        {
            "message_template_name": "confirm_message",
            "protocol": "email",
            "update_time": "2016-08-02T08:22:19Z",
            "create_time": "2016-08-02T08:22:19Z",
            "tag_names": [
                "topic_id"
            ],
            "message_template_id": "ecf63465804a4b10a0573980be78ffba"
        },
        {
            "message_template_name": "confirm_message",
            "protocol": "https",
            "update_time": "2016-08-02T08:22:20Z",
            "create_time": "2016-08-02T08:22:20Z",
            "tag_names": [
                "topic_id"
            ],
            "message_template_id": "57ba8dcecda844878c5dd5815b65d10f"
        }
    ],
    "request_id": "ce7f2f7343224f8c9597b05a9a0bcc2e",
    "message_template_count": 3
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.3.4 Querying the Details of a Message Template

Description

- API name

  QueryMessageTemplateDetail
- Function

  Query the template details, including the template content.

URI

- URI format

  GET /v2/{project_id}/notifications/message_template/{message_template_id}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
message_templa te_id	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Message Templates**.

Request

Example request

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template/
57ba8dcecda844878c5dd5815b65d10f
```

Response

- Parameter description

Parameter	Type	Description
message_template_i d	String	Template ID
message_template_ name	String	Template name

Parameter	Type	Description
protocol	String	Protocol supported by the template Currently, the following protocols are supported: ● **email** ● **default** ● **sms** ● **http** and **https**
tag_names	String	Variable list The variable name will be quoted in braces ({}) in the template. When you use a template to send messages, you can replace the variable with any content.
create_time	String	Time when the template was created The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.
update_time	String	Last time when the template was updated The UTC time is in *YYYY-MM-DDTHH:MM:SSZ* format.
content	String	Template content
request_id	String	Request ID, which is unique

- Example response

```
{
    "message_template_name": "confirm_message",
    "protocol": "https",
    "update_time": "2016-08-02T08:22:25Z",
    "create_time": "2016-08-02T08:22:20Z",
    "request_id": "ba79ca8f794f4f50985ce7b98a401b47",
    "tag_names": [
        "topic_id_id4"
    ],
    "content": "(1/24)You are invited to subscribe to topic({topic_id_id4}). Click the following URL to
confirm subscription:(If you do not want to subscribe to this topic, ignore this message.)",
    "message_template_id": "57ba8dcecda844878c5dd5815b65d10f"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.3.5 Deleting a Message Template

Description

- API name

  DeleteMessageTemplate

- Function

  Delete a message template. After you delete the template, you can no longer use it to publish messages.

URI

- URI format

  DELETE /v2/{project_id}/notifications/message_template/{message_template_id}

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
message_template_i d	Yes	String	Unique resource ID of a topic. You can obtain it by referring to **Querying Message Templates**.

Request

Example request
```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template/
b3ffa2cdda574168826316f0628f774e
```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique

- Example response
```
{
    "request_id": "5fcba32bd2814ea39431829c22bda94b"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.4 Message Publishing

5.4.1 Publishing Messages in the Text Format

Description

- API name

  Publish

- Function

  Publish messages in the text format to a topic. After the message ID is returned, the message has been saved and is to be pushed to the subscribers of the topic.

URI

- URI format

  POST /v2/{project_id}/notifications/topics/{topic_urn}/publish

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of the topic. You can obtain it by referring to **Querying Topics**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
subject	No	String	Message subject, which is used as the email subject when you publish email messages. The subject cannot exceed 512 characters.

Parameter	Mandatory	Type	Description
message	Yes	String	Message content The message content must be UTF-8-coded and can be no more than 256 KB. For SMS subscribers, if an SMS message is oversized, SMN will split it into multiple parts but only send the first two parts to subscribers. ● If the content of an SMS message consists of only ASCII characters, each SMS message can contain up to 459 characters. ● If the content includes non-ASCII characters, each SMS message can contain up to 201 characters.
time_to_live	No	String	The maximum retention period of a message in SMN After the retention period expires, SMN does not send this message. The time period is measured in seconds, and the default retention period is **3600** (one hour). The retention period must be a positive integer less than or equal to 604,800 (3600 x 24 x 7).

- Example request

  POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/urn:smn:regionId:
  f96188c7ccaf4ffba0c9aa149ab2bd57:test_create_topic_v2/publish

  ```
  {
      "subject": "test message v2",
      "message": "Message test message v2",
      "time_to_live": "3600"
  }
  ```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
message_id	String	Message ID, which is unique

- Example response

```
{
    "message_id": "bf94b63a5dfb475994d3ac34664e24f2",
    "request_id": "9974c07f6d554a6d827956acbeb4be5f"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.4.2 Publishing Messages Using a Message Structure

Description

- API name

  Publish

- Function

  Use the message structure to publish a message to a topic. After the message ID is returned, the message has been saved and is to be pushed to the subscribers of the topic. This API allows you to send different message content to different types of subscribers.

URI

- URI format

  POST /v2/{project_id}/notifications/topics/{topic_urn}/publish

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of the topic. You can obtain it by referring to **Querying Topics**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
subject	No	String	Message subject, which is presented as the email subject when SMN sends messages to email subscribers
			The message subject cannot exceed 512 bytes.
message_structure	Yes	String	Message structure, which contains JSON strings
			email, **sms**, **http**, and **https** are supported.
			The **default** protocol is mandatory. If the system fails to match any other protocols, the default message is sent.
			NOTE
			Three message formats are supported:
			● message
			● message_structure
			● message_template_name
			If the three formats are specified at the same time, they take effect in the following sequence: **message_structure** > **message_template_name** > **message**.
time_to_live	No	String	The maximum retention period of a message in SMN
			After the retention period expires, SMN does not send this message.
			Unit: second
			Default retention period: **3600** (one hour)
			The retention period must be a positive integer less than or equal to 604,800 (3600 x 24 x 7).

- Example request

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/urn:smn:regionId:
f96188c7ccaf4ffba0c9aa149ab2bd57:test_create_topic_v2/publish
{
    "subject": "test message v2",
```

```
    "time_to_live": "3600",
    "message_structure": "{\n  \"default\": \"xxx\",\n  \"APNS\": \"{\\\"aps\\\":{\\\"alert\\\":{\\\"title\\\":\
\\"xxx\\\",\\\"body\\\":\\\"xxx\\\"}}}\"\n}"
}
```

📖 **NOTE**

For example, a topic has two types of subscriptions, SMS and email. After the API is called to publish messages, the email subscriber will receive message "abc", and the SMS subscriber will receive the default message "test v2 default".

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
message_id	String	Message ID, which is unique

- Example response

```
{
    "message_id": "bf94b63a5dfb475994d3ac34664e24f2",
    "request_id": "9974c07f6d554a6d827956acbeb4be5f"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.4.3 Publishing Messages Using a Template

Description

- API name

  Publish

- Function

  Use the message template to publish a message to a topic. After the message ID is returned, the message has been saved and is to be pushed to the subscribers of the topic. This API allows you to send messages to different types of subscribers using the same template name but different protocols.

URI

- URI format

  POST /v2/{project_id}/notifications/topics/{topic_urn}/publish

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID See **Obtaining a Project ID**.
topic_urn	Yes	String	Unique resource ID of the topic. You can obtain it by referring to **Querying Topics**.

Request

- Parameter description

Parameter	Mandatory	Type	Description
subject	No	String	Message subject, which is presented as the email subject when SMN sends messages to email subscribers The message subject cannot exceed 512 bytes.
message_template_name	Yes	String	Message template name, which can be obtained by referring to **Querying Message Templates** **NOTE** Three message formats are supported: ● message ● message_structure ● message_template_name If the three formats are specified at the same time, they take effect in the following sequence: message_structure>message_template_name>message

Parameter	Mandatory	Type	Description
tags	No	Map<String,String>	Variable parameters and values **Tags** cannot be left blank. This parameter is mandatory when you use a message template to publish messages. The key in the dictionary is the parameter name in the message template and can contain a maximum of 21 characters. The value in the dictionary is the value after the key in the message template is replaced. The value cannot exceed 1 KB.
time_to_live	No	String	The maximum retention period of a message in SMN After the retention period expires, SMN does not send this message. Unit: second Default retention period: **3600** (one hour) The retention period must be a positive integer less than or equal to 604,800 (3600 x 24 x 7).

- Example request

  POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_create_topic_v2/publish

  ```
  {
      "subject": "test message template v2",
      "message_template_name": "confirm_message",
      "time_to_live": "3600",
      "tags": {
          "topic_urn": "topic_urn3331",
          "topic_id": "topic_id3332"
      }
  }
  ```

Response

- Parameter description

Parameter	Type	Description
request_id	String	Request ID, which is unique
message_id	String	Message ID, which is unique

- Example response

```
{
    "message_id": "bf94b63a5dfb475994d3ac34664e24f2",
    "request_id": "9974c07f6d554a6d827956acbeb4be5f"
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.5 Version Querying

5.5.1 Listing All SMN API Versions

Description

- API name

  QueryApiSupportVersions
- Function

  List all SMN API versions.

URI

- URI format

  GET /

Request

- Example request

  GET https://{SMN_Endpoint}/

Response

- Parameter description

Parameter	Type	Description
versions	Versions structure	Version object list. For details, see **Table 5-7**.

Table 5-7 Versions structure

Parameter	Type	Description
id	String	Version number, for example, **v2**
min_version	String	Minimum micro-version number. If the APIs do not support micro-versions, no information will be returned.
status	String	Version status, which can be the following: ● **CURRENT**: widely used version ● **SUPPORTED**: earlier version that is still supported ● **DEPRECATED**: deprecated version that may be deleted later
updated	String	Version release time, which must be UTC time. For example, the release time of v2 is 2014-06-28T12:20:21Z.
version	String	Maximum micro-version number. If the APIs do not support micro-versions, no information will be returned.
links	Links structure array	URL of an API. For details, see **Table 5-8**.

Table 5-8 Links structure

Parameter	Type	Description
href	String	Shortcut link
rel	String	Shortcut link marker name

● Example response

```
{
    "versions": [
        {
            "id": "v2",
            "links": [
                {
                    "href": "https://127.0.0.1/v2",
```

```
            "rel": "self"
        }
    ],
    "min_version": "",
    "status": "CURRENT",
    "updated": "2018-09-19T00:00:00Z",
    "version": ""
   }
  ]
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

5.5.2 Querying the SMN API v2 Version

Description

- API name

  QueryV2ApiInfo

- Function

  Query the SMN API v2 version information.

URI

- URI format

  GET /{api_version}

- Parameter description

Parameter	Mandatory	Type	Description
api_version	Yes	String	Version to be queried **NOTE** Currently, only **v2** is supported.

Request

- Example request
  ```
  GET https://{SMN_Endpoint}/v2
  ```

Response

- Parameter description

Table 5-9 Parameter in the response

Parameter	Type	Description
version	Object	Version information

Table 5-10 Description of the **version** field

Parameter	Type	Description
id	String	Version number, for example, **v2**
links	Links structure array	URL of an API. For details, see **Table 5-11**.
min_version	String	Minimum micro-version number. If the APIs do not support micro-versions, no information will be returned.
status	String	Version status, which can be the following: ● **CURRENT**: widely used version ● **SUPPORTED**: earlier version that is still supported ● **DEPRECATED**: deprecated version that may be deleted later
updated	String	Version release time, which must be UTC time. For example, the release time of v2 is 2014-06-28T12:20:21Z.
version	String	Maximum micro-version number. If the APIs do not support micro-versions, no information will be returned.

Table 5-11 Links structure

Parameter	Type	Description
href	String	Shortcut link
rel	String	Shortcut link marker name

● Example response

```
{
    "version":
        {
            "id": "v2",
            "links": [
                {
```

```
            "href": "https://127.0.0.1/v2",
            "rel": "self"
          }
        ],
        "min_version": "",
        "status": "CURRENT",
        "updated": "2018-09-19T00:00:00Z",
        "version": ""
      }
}
```

Returned Value

See **Returned Value**.

Error Codes

See **Error Codes**.

6 Public Parameters

6.1 General Request Return Code

6.1.1 Exception Response

● Parameter description

Parameter	Type	Description
request_id	String	Request ID
code	String	For details, see **Error Codes**.
message	String	For details, see **Error Codes**.

● Example

```
{
    "request_id": "aad0860d089c482b943971f802a6718e",
    "code": "SMN.0006",
    "message": "Topic not found."
}
```

6.1.2 Returned Value

● Normal

Returned Value	Description
200	The task is submitted successfully.
201	Resource creation succeeds.

● Abnormal

Returned Value	Description
400 Bad Request	Incorrect request parameters.
401 Unauthorized	Authentication failed.
403 Forbidden	No permission to access the requested resource.
404 Not Found	The requested resource does not exist.
500 Internal Server Error	The request fails because the server is abnormal.
501 Not Implemented	The request fails because the server does not support the requested function.
502 Bad Gateway	The request fails because the returned response is invalid.
503 Service Unavailable	The request fails because the system is abnormal.
504 Gateway Timeout	Gateway times out.

6.2 Error Codes

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0025	Template already exists.	The template already exists.	Create a different template.
400	SMN.0121	An application with the same name but different properties already exists.	An application with the same name but different attributes already exists.	Specify a different application name.
200	SMN.0190	This endpoint already exists.	An endpoint already exists.	Specify another endpoint.
400	SMN.0002	Parameter: Name is invalid.	Invalid **name**.	Enter a valid topic name.
400	SMN.0003	Parameter: DisplayName is invalid.	Invalid **displayname**.	Enter a valid display name for the topic.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0005	Parameter: TopicUrn is invalid.	Invalid **topic_urn**.	Specify a valid **topic_urn**.
400	SMN.0011	Parameter: Protocol is invalid.	Invalid **protocol**.	Specify a valid **protocol**.
400	SMN.0012	Parameter: Endpoint is invalid.	Invalid **endpoint**.	Specify a valid **endpoint**.
400	SMN.0014	Parameter: SubscriptionUrn is invalid.	Invalid **SubscriptionUrn**.	Specify a valid **SubscriptionUrn**.
400	SMN.0015	Parameter: Offset or limit is invalid.	Invalid **offset** or **limit**.	Enter valid **offset** and **limit** values.
400	SMN.0017	Parameter: Remark is invalid.	Invalid **remark**.	Specify a valid **remark**.
400	SMN.0019	Producer Exceptions.	Generator exception.	Contact technical support.
400	SMN.0021	MessageStructure is invalid.	Invalid **message_structure**.	Specify a valid **message_structure**.
400	SMN.0024	Parameter: content is invalid.	Invalid **content**.	Specify a valid **content**
400	SMN.00290	SubscriberUrn is unavailable.	SubscriberUrn is unavailable.	Check whether SubscriberUrn is correct.
400	SMN.0031	Parameter: template_id is invalid.	Invalid **template_id**.	Specify a valid **template_id**.
400	SMN.0032	Parameter: message_template_name is invalid.	Invalid **message_template_name**.	Specify a valid **message_template_name**.
400	SMN.0033	Parameter: push_policy is invalid.	Invalid **push_policy**.	Specify a valid **push_policy**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0034	Parameter: subscription_auth_policy is invalid.	Invalid **subscription_auth_policy**.	Specify a valid **subscription_auth_policy**.
400	SMN.0035	Parameter: validation_policy is invalid.	Invalid **validation_policy**.	Specify a valid **validation_policy**.
400	SMN.0036	Parameter: xdomain_type is invalid.	Invalid **xdomain_type**.	Specify a valid **xdomain_type**.
400	SMN.0038	Parameter: tag is invalid.	Invalid **tag**.	Specify a valid **tag**.
400	SMN.0042	Parameter: Message_template_name is invalid.	Invalid **message_template_name**.	Specify a valid **message_template_name**.
400	SMN.0043	Parameter: subscription is invalid.	Invalid **subscription**.	Specify a valid **subscription**.
400	SMN.0044	Exceeded template limit.	The maximum allowed number of templates has been reached.	A user can create a maximum of 100 message templates.
400	SMN.0046	Parameter: Attribute name is invalid.	Invalid **attribute_name**.	Specify a valid **attribute_name**.
400	SMN.0048	Parameter: Access policy is invalid.	Invalid **access_policy**.	Specify a valid **access_policy**.
400	SMN.0049	Parameter: Access policy version is invalid.	Invalid **access_policy** version.	Enter a valid version of the access policy.
400	SMN.0050	Parameter: Access policy ID is invalid.	Invalid **access_policy id**.	Specify a valid **access_policy id**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0051	Parameter: Access policy statement is invalid.	Invalid **access_policy statement**.	Specify a valid **access_policy statement**.
400	SMN.0052	Parameter: Access policy statement ID is invalid.	Invalid **access_policy statement id**.	Specify a valid **access_policy statement id**.
400	SMN.0053	Parameter: Access policy statement resource is invalid.	Invalid **access_policy statement resource**.	Specify a valid **access_policy statement resource**.
400	SMN.0054	Parameter: Access policy statement effect is invalid.	Invalid **access_policy statement effect**.	Specify a valid **access_policy statement effect**.
400	SMN.0055	Parameter: Access policy statement action is invalid.	Invalid **access_policy statement action**.	Specify a valid **access_policy statement action**.
400	SMN.0056	Parameter: Access policy statement principle is invalid.	Invalid **access_policy statement principle**.	Specify a valid **access_policy statement principle**.
400	SMN.0058	Parameter: Access policy statement condition is invalid.	Invalid **access_policy statement condition**.	Specify a valid **access_policy statement condition**.
400	SMN.0059	Parameter: job_id is invalid.	Invalid **job_id**.	Contact technical support.
400	SMN.0061	Parameter: Service name is invalid.	Invalid **Service name**.	Contact technical support.
400	SMN.0062	Parameter: Show name is invalid.	Invalid **Show name**.	Contact technical support.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0066	Parameter: System parameter name is invalid.	Invalid **System parameter name**.	Contact technical support.
400	SMN.0067	Parameter: System parameter value is invalid.	Invalid **System parameter value**.	Contact technical support.
400	SMN.0072	Parameter: Service is invalid.	Invalid **service**.	Specify a valid **service**.
400	SMN.0073	Parameter: Invalid action.	Invalid **action**.	Specify a valid **action**.
400	SMN.0074	Failed to add permission.	Failed to add permissions.	Contact technical support.
400	SMN.0077	Parameter: Status is invalid.	Invalid **status**.	Specify a valid **status**.
400	SMN.0080	The signId parameter is invalid.	Invalid **sign_id**.	Specify a valid **sign_id**.
400	SMN.0081	The signName parameter is invalid.	Invalid **sign_name**.	Specify a valid **sign_name**. The name must contain 2 to 8 characters and only letters and digits are allowed.
400	SMN.0082	The length of the remark parameter value has reached the maximum allowed.	Invalid **remark** length.	Specify a valid **remark**.
400	SMN.0087	The SMS message signature has not been approved.	SMS message signature not approved.	Specify an approved SMS message signature.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0090	Invalid domain ID.	Invalid **domain_id**.	Specify a valid **domain_id**. For details, see the section "Obtaining a Project ID".
400	SMN.0092	The kafka_id parameter is invalid.	Invalid **kafka_id**.	Contact technical support.
400	SMN.0093	The address_list parameter is invalid.	Invalid **address_list**.	Contact technical support.
400	SMN.0094	The type of queue parameter is invalid.	Invalid queue parameter type.	Contact technical support.
400	SMN.0095	The count of queue parameter is invalid.	Invalid queue parameter statistics.	Contact technical support.
400	SMN.0097	The names parameter is invalid.	Invalid **names**.	Contact technical support.
400	SMN.0100	The eventType parameter is invalid.	Invalid **event_type**.	Specify a valid **event_type**.
400	SMN.0102	Parameter: message_id is invalid.	Invalid **message_id**.	Specify a valid **message_id**.
400	SMN.0103	The start_time parameter value is invalid.	Invalid **start_time**.	Specify a valid **start_time**.
400	SMN.0104	The end_time parameter value is invalid.	Invalid **end_time**.	Specify a valid **end_time**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0108	Parameter: Type is invalid.	Invalid **type**.	Specify a valid **type**.
400	SMN.0110	Parameter: Start date is invalid.	Invalid **start_date**.	Specify a valid **start_date**.
400	SMN.0112	Parameter: End date is invalid.	Invalid **end_date**.	Specify a valid **end_date**.
400	SMN.0113	The start time and end time must be within the past 3 months.	The start time and end time must be within the past three months.	Specify the start time and end time within the past three months.
400	SMN.0114	The end time must be later than the start time.	The end time must be later than the start time.	The end time must be later than the start time.
400	SMN.0115	Parameter: Application name is invalid.	Invalid **application name**.	Specify a valid **application name**.
400	SMN.0118	Wrong Certificate.	Certificate error.	Specify a valid certificate.
400	SMN.0123	Parameter: ApplicationUrn is invalid.	Invalid **application_urn**.	Specify a valid **application_urn**.
400	SMN.0124	Parameter: application attribute is invalid.	Invalid **application_attributes**.	Specify a valid **application_attributes**.
400	SMN.0126	ContentType is wrong.	Incorrect content type.	Contact technical support.
400	SMN.0129	Certificate type is wrong.	Incorrect certificate type.	Contact technical support.
400	SMN.0132	The value of the User_data parameter is invalid.	Invalid **user_data**.	Specify a valid **user_data**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0133	The device token already exists.	The device token already exists.	Create a different device token.
400	SMN.0134	The endpoint URN is invalid.	Invalid **endpoint_urn**.	Specify a valid **endpoint_urn**.
400	SMN.0135	The endpoint does not exist.	The endpoint is not found.	Specify a valid endpoint.
400	SMN.0136	The value of the enabled parameter is invalid.	Invalid **enabled**.	Specify a valid **enabled**.
400	SMN.0137	The application URN is invalid.	Invalid **application_urn**.	Specify a valid **application_urn**.
400	SMN.0140	The topic URN configured for the message sending failure event is invalid.	Invalid **topic_urn** configured for the message sending failure event.	Specify a valid **topic_urn**.
400	SMN.0141	The topic configured for the message sending failure event does not exist.	**topic_urn** configured for the message sending failure event is not found.	Specify a valid **topic_urn**.
400	SMN.0142	The topic URN configured for the message sending success event is invalid.	Invalid **topic_urn** configured for the message sending success event.	Specify a valid **topic_urn**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0143	The topic configured for the message sending success event does not exist.	**topic_urn** configured for the message sending success event is not found.	Specify a valid **topic_urn**.
400	SMN.0144	The topic URN configured for the response message event is invalid.	Invalid **topic_urn** configured for the response message event.	Specify a valid **topic_urn**.
400	SMN.0145	The topic configured for the response message event does not exist.	**topic_urn** configured for the response message event is not found.	Specify a valid **topic_urn**.
400	SMN.0148	Parameter: user_id is invalid.	Invalid **user_id**.	Specify a valid **user_id**
400	SMN.0150	Complete real-name registration using your company's certificate.	Complete real-name registration using your company's certificate.	Contact technical support.
400	SMN.0151	The message signature type cannot be promotional.	The signature cannot be a promotional SMS message signature.	Specify a transactional SMS message signature.
400	SMN.0152	The number allocated to the SMS template has reached the maximum allowed.	The maximum allowed number of SMS templates was reached.	Create no more than 100 SMS templates.
400	SMN.0155	Parameter: SmsTemplateType is invalid.	Invalid **sms_template_type**.	Specify a valid **sms_template_type**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0156	Parameter: SmsTemplate Name is invalid.	Invalid **sms_template_name**.	Specify a valid **sms_template_name**.
400	SMN.0157	Parameter: SmsTemplate Content is invalid.	Invalid **sms_template_content**.	Specify a valid **sms_template_content**.
400	SMN.0159	The number of endpoints has reached the maximum allowed.	Endpoints reached the quota.	Enter no more than 1,000 phone numbers.
400	SMN.0160	Invalid endpoint.	Invalid endpoint.	Specify a valid endpoint.
400	SMN.0161	Invalid SMS template ID.	Invalid **sms_template_id**.	Specify a valid **sms_template_id**.
400	SMN.0188	Invalid SMS template type.	Invalid SMS template type.	The template must be a promotional SMS template.
400	SMN.0189	Failed to obtain the certificate required for reviewing the SMS message template function.	Failed to obtain the certificate required for viewing the SMS message template.	Contact technical support.
400	SMN.0191	Parameter: Invalid enterprise name.	Invalid enterprise name.	Contact technical support.
400	SMN.0192	Parameter: Invalid enterprise certificate ID.	Invalid enterprise certificate ID.	Contact technical support.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0204	Promotional SMS messages must contain the way to unsubscribe in the end, for example, by replying "TD", or "T".	Promotional SMS messages must include content at the end explaining how recipients can unsubscribe, for example, by replying "unsubscribe".	Promotional SMS messages must include content at the end explaining how recipients can unsubscribe, for example, by replying "unsubscribe".
400	SMN.0215	The SMS message signature ID does not match the signature name in the SMS message template content.	The SMS message signature ID does not match the signature name in **sms_template_content**.	Specify a signature ID that matches the signature name in **sms_template_content**.
400	SMN.0216	The template content does not include any valid SMS message signature.	No valid SMS message signature in the template **Content**.	Specify a valid signature in the template **Content**.
400	SMN.0217	The template content does not include any SMS message signature.	No SMS message signature in **sms_template_content**.	**sms_template_content** must contain an approved and available signature, for example, [XX mall].
400	SMN.0224	The SMS message template is invalid because the associated signature has been deleted.	The SMS template has expired due to the deletion of the associated signature.	Specify a valid **sms_template_id**.

Status Code	Error Code	Error Message	Description	Solution
400	SMN.0227	The message contains sensitive words.	Sensitive words contained in the message content.	Modify the message content.
400	SMN.0230	Parameter: Invalid title.	Invalid **title**.	Specify a valid **title**.
400	SMN.0261	Check whether your account balance is sufficient and top up your account if necessary.	Your account balance may be insufficient.	Check the account balance and top up your account if necessary.
403	SMN.0001	No permission to request resources.	No permissions for the requested resource.	Request the required permissions.
403	SMN.0004	Exceeded topic limit.	Topics exceeded the maximum number allowed.	Create no more than 3,000 topics.
403	SMN.0007	Exceeded subscription limit.	Subscriptions exceeded the maximum number allowed.	Add no more than 10,000 subscriptions to a topic.
403	SMN.0008	Parameter: Subject is invalid.	Invalid **subject**.	Specify a valid **subject**.
403	SMN.0009	Parameter: Message is invalid.	Invalid **message**.	Specify a valid **message**.
403	SMN.0022	Parameter: token is invalid.	Invalid **token**.	Specify a valid **token**.
403	SMN.00308	Policy doesn't allow {} to be performed.	The policy cannot be executed.	Check the SMN permissions.

Status Code	Error Code	Error Message	Description	Solution
403	SMN.0047	Parameter: Value exceeds the maximum length.	**value** exceeds the maximum length.	Specify a valid **value**.
403	SMN.0057	The number of SMS messages exceeds the limit.	The maximum number of SMS messages has been reached.	Send no more than 60 SMS messages to a phone number within 1 hour and no more than 200 within 12 hours.
403	SMN.0065	The number of email messages exceeds the limit.	The maximum number of email messages has been reached.	Contact technical support.
403	SMN.0069	Not authorized to subscribe internal endpoints.	Subscription endpoint not authorized.	Specify a correct HTTP/HTTPS endpoint.
403	SMN.0070	No permission to request resources. The role is op_restricted. Contact customer service.	No permissions for the requested resource. Operation permissions limited.	Check the account balance and top up your account if necessary. If the fault persists, contact customer service.
403	SMN.0071	No permission to request resources. The role is op_suspended.	No permissions for the requested resource. Your role is suspended.	Check whether your account has been frozen.
403	SMN.0075	Parameter: tags are too many.	The maximum allowed number of tags has been reached.	Enter no more than 90 non-repeated variables in the message template.

Status Code	Error Code	Error Message	Description	Solution
403	SMN.0078	The maximum number of subscriptions has been reached.	The maximum allowed number of subscriptions has been reached.	Create a topic.
403	SMN.0083	Finish real-name authentication before continuing.	The real-name authentication is not completed.	Contact technical support.
403	SMN.0086	The number allocated to the SMS message signature has reached the maximum allowed.	The maximum allowed number of SMS message signatures has been reached.	Create no more than 10 SMS message signatures.
403	SMN.0099	The SMN service has not been authorized to send messages to this queue.	SMN is not authorized to send messages to this queue.	Grant permissions to SMN in the queue policy.
403	SMN.0116	Platform not supported.	Unsupported platform.	Specify a supported platform.
403	SMN.0117	Certificate type is not supported.	Unsupported certificate type.	Specify a supported certificate type.
403	SMN.0119	PlatformPrinciple must be at most 8192 bytes long in UTF-8 encoding.	UTF-8 encoded **platform_principle** exceeded 8,192 bytes.	Specify a valid, UTF-8 encoded value for **platform_principle** that does not exceed 8,192 bytes.

Status Code	Error Code	Error Message	Description	Solution
403	SMN.0120	PlatformCredential must be at most 8192 bytes long in UTF-8 encoding.	UTF-8 encoded **platform_credential** must not exceed 8,192 bytes.	Specify a valid, UTF-8 encoded value for **platform_credential** that does not exceed 8,192 bytes.
403	SMN.0127	File is too large.	Excessive file size.	Contact technical support.
403	SMN.0128	Password is empty.	Password not specified.	Contact technical support.
403	SMN.0131	The value of the Token parameter is invalid.	Invalid **token**.	Specify a valid **token**.
403	SMN.0147	Only enterprise users who use the certificate images for authentication can create message signatures.	Only enterprise users who use certificates for identity authentication can create message signatures.	Contact technical support.
403	SMN.0153	Only enterprise users can create SMS message Template.	Only enterprise users can create SMS templates.	Complete enterprise authentication.
403	SMN.0154	Only enterprise users who use the certificate images for authentication can create sms message templates.	Only enterprise users who use certificates for identity authentication can create SMS templates.	Contact technical support.
403	SMN.0165	Parameter: tenant_token is invalid.	Invalid **tenant_token**.	Specify a valid **tenant_token**.

Status Code	Error Code	Error Message	Description	Solution
403	SMN.0195	Complete enterprise real-name authentication first.	The real-name authentication is not completed.	Complete real-name authentication.
403	SMN.0228	Do not have permission to use this API. Contact customer service to apply for permission.	No permissions to use this API.	Contact technical support to apply for permissions required to use this API.
404	SMN.0006	Topic not found.	Topic not found.	Create a topic.
404	SMN.0013	Subscription resource not found.	No subscription endpoint found.	Specify a valid subscription endpoint.
404	SMN.0027	Template not found.	Template not found.	Use an existing template.
404	SMN.0045	Attribute not found.	Attribute not found.	Specify a valid attribute.
404	SMN.0060	Job not found.	Failed to query the job.	Contact technical support.
404	SMN.0063	Cloud service not found.	Cloud service not found.	Contact technical support.
404	SMN.0068	System parameter not found.	System parameter not found.	Contact technical support.
404	SMN.0076	Default message template not found.	The message template using the **Default** protocol is not found.	Create a message template using the **Default** protocol.
404	SMN.0084	The SMS message signature could not be found.	SMS message signature not found.	Specify an approved SMS message signature.

Status Code	Error Code	Error Message	Description	Solution
404	SMN.0091	Default system message template not found.	The message template using the **Default** protocol is not found.	Create a system template using the **Default** protocol.
404	SMN.0096	Kafka not found.	Kafka not found.	Contact technical support.
404	SMN.0098	This queue does not exist.	The queue is not found.	Specify a valid DMS queue.
404	SMN.0101	The graph does not exist in the functiongraph service.	The function does not exist in FunctionGraph.	Check whether the FunctionGraph endpoint is valid.
404	SMN.0109	Start date is not found.	**start_date** not found.	**start_date** must be specified.
404	SMN.0111	End date is not found.	**end_date** not found.	**end_date** must be specified.
404	SMN.0122	Application not found.	Application not found.	Specify a different application name.
404	SMN.0125	Endpoint does not exist.	The endpoint is not found.	Specify a valid application endpoint.
404	SMN.0130	The application does not exist.	The application platform is not found.	Specify a valid **application_urn**.
404	SMN.0162	SMS template not found.	The SMS template is not found.	Specify a valid SMS template.
404	SMN.0167	Phone number does not exist.	The phone number is not found.	Specify a valid phone number.
404	SMN.0168	Email Address does not exist.	The email address is not found.	Specify a valid email address.
500	SMN.0016	Database Exceptions.	Database exception.	Contact technical support.

Status Code	Error Code	Error Message	Description	Solution
500	SMN.0018	Service internal error.	Internal service error.	Contact technical support.
500	SMN.0089	An error occurred in the BSS system. Try again later.	BSS error. Try again later.	Contact technical support.
500	SMN.0158	Failed to upload the file.	Failed to upload the file.	Contact technical support.

6.3 Obtaining a Project ID

A project ID is required for some URLs when an API is called. Therefore, you need to obtain a project ID in advance. The steps are as follows:

1. Obtain the token.

   For details, see **Token Authentication**.

2. Obtain a project ID.

   The API for obtaining the project ID is **GET https://iam.eu-west-0.myhuaweicloud.com/v3/projects**.

   Add **X-Auth-Token** to the request header and set its value to the token obtained in the preceding step.

   The following is an example response. The value of **id** is the project ID to be obtained.

```
{
    "links": {},
    "projects": [
        {
            "is_domain": ,
            "description": "",
            "links": {},
            "enabled": true,
            "id": "",   // Project ID
            "parent_id": "",
            "domain_id": "",
            "name": ""
        },
        …
    ]
}
```

A Appendix

A.1 Topic Attribute List

Attribute Name	Function	Default Value
access_policy	Configures topic access policy, which is used to limit the access permission of other users.	Empty string
introduction	Configures the topic introduction, which is displayed in the email or SMS confirmation messages.	Empty string