

消息通知服务

接口参考

文档版本 01
发布日期 2025-03-04



版权所有 © 华为技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <https://www.huawei.com>

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 使用前必读	1
2 API 概览	3
3 如何调用 API	4
3.1 构造请求	4
3.2 认证鉴权	7
3.3 返回结果	9
4 快速入门	11
5 API	13
5.1 主题操作	13
5.1.1 查询主题列表	13
5.1.2 创建主题	17
5.1.3 查询主题详情	23
5.1.4 更新主题	26
5.1.5 删除主题	31
5.1.6 查询主题策略	34
5.1.7 删除所有主题策略	37
5.1.8 更新主题策略	40
5.1.9 删除指定名称的主题策略	48
5.2 订阅操作	51
5.2.1 查询订阅者列表	51
5.2.2 查询指定 Topic 的订阅者列表	56
5.2.3 订阅	60
5.2.4 取消订阅	72
5.2.5 更新订阅者	74
5.3 模板操作	80
5.3.1 查询消息模板列表	80
5.3.2 创建消息模板	84
5.3.3 查询消息模板详情	90
5.3.4 更新消息模板	93
5.3.5 删除消息模板	98
5.4 发布消息操作	101
5.4.1 消息发布	101

5.4.2 发布探测消息.....	117
5.4.3 获取 http 探测结果.....	124
5.5 使用标签管理服务.....	130
5.5.1 查询资源实例.....	130
5.5.2 批量添加删除资源标签.....	146
5.5.3 查询资源标签.....	156
5.5.4 添加资源标签.....	159
5.5.5 查询项目标签.....	165
5.5.6 删除资源标签.....	169
5.6 查询版本操作.....	171
5.6.1 查询 SMN 支持的 API 版本号信息.....	171
5.6.2 查询 SMN API V2 版本信息.....	175
5.7 云日志操作.....	178
5.7.1 绑定云日志.....	178
5.7.2 查询云日志.....	183
5.7.3 更新云日志.....	187
5.7.4 解绑云日志.....	192
5.8 订阅过滤策略操作.....	195
5.8.1 批量创建订阅过滤策略.....	195
5.8.2 批量更新订阅过滤策略.....	203
5.8.3 批量删除订阅过滤策略.....	210
5.9 通知策略.....	216
5.9.1 创建通知策略.....	216
5.9.2 查询通知策略.....	223
5.9.3 修改通知策略.....	228
5.9.4 删除通知策略.....	235
6 应用示例.....	241
6.1 示例 1: 使用主题发布消息.....	241
6.2 示例 2: 使用消息模板方式发布消息.....	242
7 权限和授权项.....	245
7.1 权限及授权项说明.....	245
7.2 主题操作.....	246
7.3 模板操作.....	247
7.4 标签操作.....	248
7.5 发布消息操作.....	248
7.6 Application 操作 (废弃)	249
7.7 Application 直发消息操作 (废弃)	249
8 公共参数.....	250
8.1 获取项目 ID.....	250
8.2 application 消息体格式 (废弃)	251
9 历史 API.....	253

9.1 Application 操作.....	253
9.1.1 查询 Application (废弃)	253
9.1.2 创建 Application (废弃)	257
9.1.3 查询 Application 属性 (废弃)	261
9.1.4 更新 Application (废弃)	263
9.1.5 删除 Application (废弃)	266
9.2 Application endpoint 操作.....	269
9.2.1 查询 Application 的 Endpoint 列表 (废弃)	269
9.2.2 创建 Application endpoint (废弃)	273
9.2.3 查询 Application 的 Endpoint 属性 (废弃)	276
9.2.4 更新 Application endpoint (废弃)	279
9.2.5 删除 Application endpoint (废弃)	282
9.3 Application 直发消息操作.....	284
9.3.1 App 消息发布 (废弃)	284
9.4 订阅确认 V2 接口 (废弃)	290
A 附录.....	292
A.1 Topic 属性表.....	292
A.2 通用请求返回值.....	292
A.2.1 异常响应.....	292
A.2.2 状态码.....	293
A.3 错误码.....	293
A.4 国家码和地区码.....	309

1 使用前必读

欢迎使用消息通知服务（Simple Message Notification, SMN）。消息通知服务是可靠的、可扩展的、海量的消息处理服务。它可以依据您的需求主动推送通知消息，最终您可以通过短信、电子邮件、应用等方式接收。您也可以应用之间通过SMN实现应用的功能集成，降低系统的复杂性。

您可以使用本文档提供的API对消息通知服务进行相关操作，如创建、查询、删除、更新等。支持的全部操作请参见[2 API概览](#)。

在调用消息通知服务API之前，请确保已经充分了解消息通知服务相关概念，详细信息请参见《消息通知服务用户指南》的“[产品介绍](#)”。

终端节点

终端节点（Endpoint）即调用API的[请求地址](#)，不同服务不同区域的终端节点不同，您可以从[地区和终端节点](#)中查询所有服务的终端节点。

基本概念

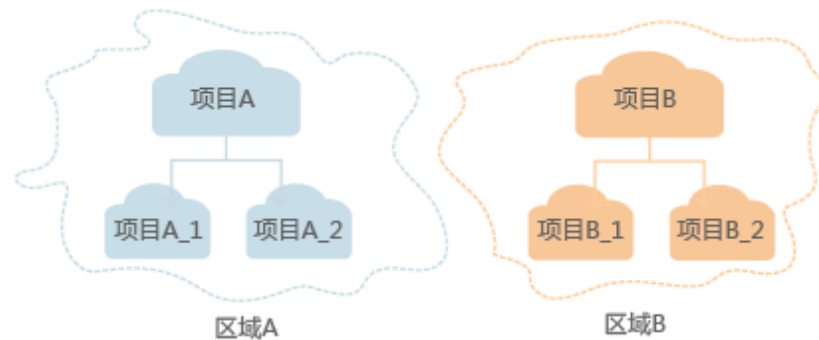
- 账号
用户注册时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用用户进行日常管理工作。
- 用户
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域（Region）
从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。
详情请参见[区域和可用区](#)。
- 可用区（AZ, Availability Zone）

一个可用区是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

- 项目

区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中创建资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中的资源，使得资源的权限控制更加精确。

图 1-1 项目隔离模型



- 企业项目

企业项目是项目的升级版，针对企业不同项目间的资源进行分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。

关于企业项目ID的获取及企业项目特性的详细信息，请参见《[企业管理用户指南](#)》。

2 API 概览

通过使用消息通知服务所提供的接口，您可以完整的使用消息通知服务的所有功能。例如创建Topic、添加订阅。

表 2-1 接口说明

类型	说明
主题操作	包含创建主题、更新主题、删除主题等接口。
订阅操作	包含查询订阅列表、订阅、取消订阅等接口。
模板操作	包含创建消息模板、更新消息模板、删除消息模板等接口。
发布消息操作	包含消息发布、发布探测消息、获取http探测结果等接口。
使用标签管理服务	包含查询资源实例、添加资源标签和查询项目标签等接口。
查询版本操作	包含查询SMN支持的API版本号信息和查询SMN API V2版本信息接口。
云日志操作	包含绑定云日志、查询云日志、更新云日志等接口。
订阅过滤策略操作	包含批量创建订阅过滤策略、批量更新订阅过滤策略、批量删除订阅过滤策略接口。
通知策略	包含创建通知策略、查询通知策略、修改通知策略、删除通知策略接口。

3 如何调用 API

3.1 构造请求

3.2 认证鉴权

3.3 返回结果

3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM服务的[管理员创建IAM用户](#)来说明如何调用API，该API获取用户的Token，Token可以用于调用其他API时鉴权。

请求 URI

请求URI由如下部分组成：

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

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

表 3-1 URI 中的参数说明

参数	描述
URI-scheme	表示用于传输请求的协议，当前所有API均采用HTTPS协议。
Endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从 地区和终端节点 获取。 例如IAM服务在“中国-香港”区域的Endpoint为“iam.ap-southeast-1.myhuaweicloud.com”。
resource-path	资源路径，即API访问路径。从具体API的URI模块获取，例如“获取用户Token”API的resource-path为“/v3/auth/tokens”。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“？limit=10”，表示查询不超过10条数据。

例如您需要您创建IAM用户，由于IAM为全局服务，则使用任一区域的Endpoint，比如“中国-香港”区域的Endpoint（iam.ap-southeast-1.myhuaweicloud.com），并在[管理员创建IAM用户](#)的URI部分找到resource-path（/v3.0/OS-USER/users），拼接起来如下所示。

`https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users`

图 3-1 URI 示意图



说明

为方便查看，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务您正在请求什么类型的操作。

表 3-2 HTTP 方法

方法	说明
GET	请求服务器返回指定资源。
PUT	请求服务器更新指定资源。
POST	请求服务器新增资源或执行特殊操作。
DELETE	请求服务器删除指定资源，如删除对象等。
HEAD	请求服务器资源头部。
PATCH	请求服务器更新资源的部分内容。 当资源不存在的时候，PATCH可能会去创建一个新的资源。

在[管理员创建IAM用户](#)的URI部分，您可以看到其请求方法为“POST”，则其请求为：

`POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users`

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

详细的公共请求消息头字段请参见[表3-3](#)。

表 3-3 公共请求消息头

名称	描述	是否必选	示例
Host	请求的服务器信息，从服务API的URL中获取。值为hostname[:port]。端口缺省时使用默认的端口，https的默认端口为443。	否 使用AK/SK认证时该字段必选。	code.test.com or code.test.com:443
Content-Type	消息体的类型（格式）。推荐用户使用默认值application/json，有其他取值时会在具体接口中专门说明。	是	application/json
Content-Length	请求body长度，单位为Byte。	否	3495
X-Project-Id	project id，项目编号。请参考 8.1 获取项目ID 章节获取项目编号。	否 如果是专属云场景采用AK/SK认证方式的接口请求，或者多project场景采用AK/SK认证的接口请求，则该字段必选。	e9993fc787d94b6c886cb aa340f9c0f4
X-Auth-Token	用户Token。 用户Token也就是调用 获取用户Token 接口的响应值，该接口是唯一不需要认证的接口。 请求响应成功后在响应消息头（Headers）中包含的“X-Subject-Token”的值即为Token值。	否 使用Token认证时该字段必选。	注：以下仅为Token示例片段。 MIIPAgYJKoZlhvcNAQcCo ...ggg1BBIIlNPXsidG9rZ

📖 说明

API同时支持使用AK/SK认证，AK/SK认证使用SDK对请求进行签名，签名过程会自动往请求中添加Authorization（签名认证信息）和X-Sdk-Date（请求发送的时间）请求头。

AK/SK认证的详细说明请参见[3.2 认证鉴权](#)的“AK/SK认证”。

对于[管理员创建IAM用户](#)接口，使用AK/SK方式认证时，添加消息头后的请求如下所示。

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****
```

请求消息体（可选）

该部分可选。请求消息体通常以结构化格式（如JSON或XML）发出，与请求消息头中Content-Type对应，传递除请求消息头之外的内容。若请求消息体中的参数支持中文，则中文字符必须为UTF-8编码。

每个接口的请求消息体内容不同，也并不是每个接口都需要有请求消息体（或者说消息体为空），GET、DELETE操作类型的接口就不需要消息体，消息体具体内容需要根据具体接口而定。

对于[管理员创建IAM用户](#)接口，您可以从接口的请求部分看到所需的请求参数及参数说明，将消息体加入后的请求如下所示，其中加粗的字段需要根据实际值填写。

- **accountid**为IAM用户所属的账号ID。
- **username**为要创建的IAM用户名。
- **email**为IAM用户的邮箱。
- *********为IAM用户的登录密码。

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****

{
  "user": {
    "domain_id": "accountid",
    "name": "username",
    "password": "*****",
    "email": "email",
    "description": "IAM User Description"
  }
}
```

到这里为止这个请求需要的内容就具备齐全了，您可以使用[curl](#)、[Postman](#)或直接编写代码等方式发送请求调用API。对于获取用户Token接口，返回的响应消息头中的“X-Subject-Token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key）加密调用请求。推荐使用AK/SK认证，其安全性比Token认证要高。

- Token认证：通过Token认证调用请求。

AK/SK 认证

📖 说明

AK/SK签名认证方式仅支持消息体大小在12MB以内，12MB以上的请求请使用Token认证。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK (Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK (Secret Access Key)：私有访问密钥。与访问密钥ID结合使用，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。详细的签名方法和SDK使用方法请参见[API签名指南](#)。

📖 说明

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

Token 认证

📖 说明

Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头中，从而通过身份认证，获得操作API的权限。Token可通过调用[获取用户Token](#)接口获取。

调用本服务API需要项目级别的Token，即调用[获取用户Token](#)接口时，请求body中auth.scope的取值需要选择project，如下所示。

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username", //IAM用户名
          "password": $ADMIN_PASS, //IAM用户密码，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全
          "domain": {
            "name": "domainname" //IAM用户所属账号名
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxx" //项目名称
      }
    }
  }
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFJ...”，则调用接口时将“X-Auth-Token: ABCDEFJ...”加到请求消息头即可，如下所示。

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

3.3 返回结果

状态码

请求发送以后，您会收到响应，其中包含状态码、响应消息头和消息体。

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见[A.2.2 状态码](#)。

对于[管理员创建IAM用户](#)接口，如果调用后返回状态码为“201”，则表示请求成功。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于[管理员创建IAM用户](#)接口，返回如[图3-2](#)所示的消息头，其中“X-Subject-Token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

说明

建议在配置文件或者环境变量中密文存放，使用时解密，确保安全。

图 3-2 管理员创建 IAM 用户响应消息头

```
"X-Frame-Options": "SAMEORIGIN",
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fceac8a",
"Transfer-Encoding": "chunked",
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",
"Server": "api-gateway",
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",
"X-Content-Type-Options": "nosniff",
"Connection": "keep-alive",
"X-Download-Options": "noopen",
"X-XSS-Protection": "1; mode=block;",
"X-IAM-Trace-Id": "token_█_null_af2953f2bcc67a42325a69a19e6c32a2",
"Date": "Tue, 21 May 2024 09:03:40 GMT",
"Content-Type": "application/json; charset=utf8"
```

响应消息体（可选）

该部分可选。响应消息体通常以结构化格式（如JSON或XML）返回，与响应消息头中Content-Type对应，传递除响应消息头之外的内容。

对于[管理员创建IAM用户](#)接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "user": {
```

```
"id": "c131886aec...",  
"name": "IAMUser",  
"description": "IAM User Description",  
"areacode": "",  
"phone": "",  
"email": "****@***.com",  
"status": null,  
"enabled": true,  
"pwd_status": false,  
"access_mode": "default",  
"is_domain_owner": false,  
"xuser_id": "",  
"xuser_type": "",  
"password_expires_at": null,  
"create_time": "2024-05-21T09:03:41.000000",  
"domain_id": "d78cbac1.....",  
"xdomain_id": "30086000.....",  
"xdomain_type": "",  
"default_project_id": null  
}  
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

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

其中，error_code表示错误码，error_msg表示错误描述信息。

4 快速入门

操作场景

本节通过调用一系列SMN的API实现通过主题发布消息。

API的调用方法请参见[3 如何调用API](#)。

📖 说明

通过IAM服务获取到的Token有效期为24小时，需要使用同一个Token鉴权时，可以先将Token缓存，避免频繁调用。

涉及 API

当您使用Token认证方式完成认证鉴权时，需要获取用户Token并在调用接口时增加“X-Auth-Token”到业务接口请求消息头中。

- IAM获取token的API
- SMN创建Topic的API
- SMN添加订阅的API
- SMN发送消息的API

具体步骤

1. Token认证，具体操作请参考[3.2 认证鉴权](#)。
2. 创建Topic。发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics”。
 - a. 在Request Header中增加“X-Auth-Token”。
 - b. 在Request Body中传入参数如下：

```
{
  "name": "test_topic_v2", //Topic名称（必填String）
  "display_name": "test" //显示名（非必填String）
}
```

请求响应成功后，返回request_id和topic_urn。
如果请求失败，则会返回错误码及对应的错误信息说明，详细错误码信息请参考[错误码](#)。
3. 添加订阅。发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/subscriptions”。

其中，topic_urn可在**2.b**的返回体中获取。

- a. 在Request Header中增加“X-Auth-Token”。
- b. 在Request Body中传入参数如下：

```
{  
  "protocol": "email", //订阅协议（必填String）  
  "endpoint": "xxx@xx.com", //邮件地址（必填String）  
  "remark": "订阅终端" //备注（非必填String）  
}
```

请求响应成功后，返回request_id和subscription_urn。

如果请求失败，则会返回错误码及对应的错误信息说明，详细错误码信息请参考[错误码](#)。

说明

可为Topic添加多种以及多个订阅终端，支持的订阅协议请参考[订阅](#)。

- c. 订阅终端会收到一条确认订阅的消息，请在邮箱中进行确认订阅的操作。
4. 发送消息。发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/publish”。

- a. 在Request Header中增加“X-Auth-Token”。
- b. 在Request Body中传入参数如下：

```
{  
  "subject": "test message v2", //标题，邮件订阅终端使用（非必填String）  
  "message": "Message test message v2." //需要发送的消息（必填String）  
}
```

请求响应成功后，返回request_id和message_id。

如果请求失败，则会返回错误码及对应的错误信息说明，详细错误码信息请参考[错误码](#)。

说明

已完成**3.c**中的确认订阅操作的订阅终端都能收到该消息，否则无法接收消息。

5 API

- 5.1 主题操作
- 5.2 订阅操作
- 5.3 模板操作
- 5.4 发布消息操作
- 5.5 使用标签管理服务
- 5.6 查询版本操作
- 5.7 云日志操作
- 5.8 订阅过滤策略操作
- 5.9 通知策略

5.1 主题操作

5.1.1 查询主题列表

功能介绍

分页查询Topic列表，Topic列表按照Topic创建时间进行降序排列。分页查询可以指定offset以及limit。如果不存在Topic，则返回空列表。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/notifications/topics

表 5-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

表 5-2 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量。 偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。
limit	否	Integer	查询的数量限制。 取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。
enterprise_project_id	否	String	企业项目id。
name	否	String	检索的主题名称，完全匹配。
fuzzy_name	否	String	检索的主题名称，模糊匹配。
topic_id	否	String	检索的主题ID，完全匹配。
fuzzy_display_name	否	String	检索的主题显示名。模糊匹配。 参数字节长度不能大于192字节。

请求参数

表 5-3 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-4 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
topic_count	Integer	返回的Topic个数。该参数不受offset和limit影响，即返回的是您账户下所有的Topic个数。
topics	Array of ListTopicsItem objects	Topic结构体数组。

表 5-5 ListTopicsItem

参数	参数类型	描述
topic_urn	String	Topic的唯一的资源标识。
name	String	创建topic的名字。
display_name	String	Topic的显示名，推送邮件消息时，作为邮件发件人显示。
push_policy	Integer	消息推送的策略，该属性目前不支持修改，后续将支持修改。0表示发送失败，保留到失败队列，1表示直接丢弃发送失败的消息。
enterprise_project_id	String	企业项目ID。
topic_id	String	主题ID。

状态码：400

表 5-6 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-7 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-8 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-9 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询主题列表

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

响应示例

状态码：200

OK

```
{  
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
```

```
"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",
  "topic_id": "84698185946d44d08431aa441d8e2cf2"
}]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.2 创建主题

功能介绍

创建Topic，单用户默认配额为3000。高并发场景下，可能会出现Topic数量超过3000仍创建成功的情况，此为正常现象。

接口是幂等的，接口调用返回成功时，若已存在同名的Topic，返回的status code为200，否则返回的status code为201

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/notifications/topics

表 5-10 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

请求参数

表 5-11 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-12 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	创建topic的名字。Topic名称只能包含大写字母、小写字母、数字、-和_，且必须由大写字母、小写字母或数字开头，长度为1到255个字符。
display_name	是	String	Topic的显示名，推送邮件消息时，作为邮件发件人显示。显示名的长度为192byte或64个中文。默认值为空。
enterprise_project_id	否	String	企业项目ID。非必选参数，当企业项目开关打开时需要传入该参数。

响应参数

状态码：201

表 5-13 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
topic_urn	String	Topic的唯一的资源标识，可通过查询主题列表获取该标识。

状态码：400

表 5-14 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-15 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-16 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-17 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

创建名称为test_topic_v2，且显示名为testtest的主题

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics
```

```
{
  "name" : "test_topic_v2",
  "display_name" : "testtest"
}
```

响应示例

状态码：201

OK

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

SDK 代码示例

SDK代码示例如下。

Java

创建名称为test_topic_v2，且显示名为testtest的主题

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class CreateTopicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateTopicRequest request = new CreateTopicRequest();
        CreateTopicRequestBody body = new CreateTopicRequestBody();
        body.withDisplayName("testtest");
        body.withName("test_topic_v2");
        request.withBody(body);
        try {
            CreateTopicResponse response = client.createTopic(request);
        }
    }
}
```

```
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

创建名称为test_topic_v2，且显示名为testtest的主题

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateTopicRequest()
        request.body = CreateTopicRequestBody(
            display_name="testtest",
            name="test_topic_v2"
        )
        response = client.create_topic(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建名称为test_topic_v2，且显示名为testtest的主题

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateTopicRequest{}
    request.Body = &model.CreateTopicRequestBody{
        DisplayName: "testtest",
        Name: "test_topic_v2",
    }
    response, err := client.CreateTopic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.3 查询主题详情

功能介绍

查询Topic的详细信息。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-18 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-19 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-20 响应 Body 参数

参数	参数类型	描述
update_time	String	更新时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
push_policy	Integer	消息推送的策略。0表示发送失败，保留到失败队列，1表示直接丢弃发送失败的消息。

参数	参数类型	描述
create_time	String	创建时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
name	String	创建Topic的名字。
topic_urn	String	Topic的唯一的资源标识。可以通过 查看主题列表 获取该标识。
display_name	String	Topic的显示名，推送邮件消息时，作为邮件发件人显示。
request_id	String	请求的唯一标识ID。
enterprise_project_id	String	企业项目ID。
topic_id	String	主题ID。

状态码：400

表 5-21 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-22 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-23 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-24 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询主题详情

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

响应示例

状态码：200

OK

```
{
  "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",
  "topic_id" : "84698185946d44d08431aa441d8e2cf2"
}
```

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.4 更新主题

功能介绍

更新显示名。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-25 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识。可以通过 查看主题列表 获取该标识。

请求参数

表 5-26 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-27 请求 Body 参数

参数	是否必选	参数类型	描述
display_name	是	String	Topic的显示名，推送邮件消息时，作为邮件发件人显示。显示名的长度为192byte或64个中文。

响应参数

状态码：200

表 5-28 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-29 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-30 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-31 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-32 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新名称为test_topic_v2的主题的显示名为testtest222

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

响应示例

状态码：200

OK

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

SDK 代码示例

SDK代码示例如下。

Java

更新名称为test_topic_v2的主题的显示名为testtest222

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class UpdateTopicSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateTopicRequest request = new UpdateTopicRequest();
        request.withTopicUrn("{topic_urn}");
        UpdateTopicRequestBody body = new UpdateTopicRequestBody();
        body.withDisplayName("testtest222");
        request.withBody(body);
        try {
            UpdateTopicResponse response = client.updateTopic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新名称为test_topic_v2的主题的显示名为testtest222

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
```

```
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = SmnClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(SmnRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateTopicRequest()
    request.topic_urn = "{topic_urn}"
    request.body = UpdateTopicRequestBody(
        display_name="testtest222"
    )
    response = client.update_topic(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

更新名称为test_topic_v2的主题的显示名为testtest222

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateTopicRequest{}
    request.TopicUrn = "{topic_urn}"
    request.Body = &model.UpdateTopicRequestBody{
        DisplayName: "testtest222",
    }
    response, err := client.UpdateTopic(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
```

```
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.5 删除主题

功能介绍

删除主题。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-33 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-34 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-35 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID

状态码：400

表 5-36 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-37 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-38 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-39 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除主题

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

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.6 查询主题策略

功能介绍

查询主题的策略信息。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-40 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

表 5-41 Query 参数

参数	是否必选	参数类型	描述
name	否	String	主题策略名称。只支持特定的策略名称，请参见 Topic属性表 。

请求参数

表 5-42 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-43 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
attributes	TopicAttribute object	属性的键值对。access_policy: topic的访问策略。introduction: topic的简介。

表 5-44 TopicAttribute

参数	参数类型	描述
access_policy	String	topic的访问策略
introduction	String	topic的简介

状态码：400

表 5-45 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-46 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-47 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-48 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询主题策略

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

响应示例

状态码：200

OK

```
{
  "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\": \"__org_path_pub_0\", \"Effect\": \"Allow\", \"Principal\": {\"OrgPath\": [\"ou-bf966fe82ebb4d35d68b791729228788/r-001ebf32880a13eabfc8e1c37eee3ae9/ou-0dbfffe92fd92ddb35feff9b4079459c\"]}, \"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\"} ]}"
  }
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.7 删除所有主题策略

功能介绍

删除所有主题策略。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-49 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-50 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-51 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-52 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-53 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-54 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-55 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除所有主题策略

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/attributes
null
```

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.8 更新主题策略

功能介绍

更新主题的策略信息。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-56 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
name	是	String	主题策略名称。只支持特定的策略名称，请参见 Topic属性表 。

请求参数

表 5-57 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-58 请求 Body 参数

参数	是否必选	参数类型	描述
value	是	String	1. 当请求参数name为 access_policy时, value为 Topic属性值, 最大支持 30KB。 2. 当请求参数name为 introduction时, value为 topic简介, 最大长度120B。

响应参数

状态码：200

表 5-59 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-60 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-61 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-62 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-63 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

- 更新主题策略

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\": \"__org_path_pub_0\", \"Effect\": \"Allow\", \"Principal\": { \"OrgPath\": [ \"o-bf966fe82ebb4d35d68b791729228788/r-001ebf32880a13eabfc8e1c37eee3ae9/ou-0dbfffe92fd92ddb35feff9b4079459c\" ] }, \"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}\" ] } ] }"
```

- 更新主题策略

PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/attributes/introduction

```
{
  "value" : ""
}
```

响应示例

状态码：200

OK

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


SDK 代码示例

SDK代码示例如下。

Java

- 更新主题策略

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class UpdateTopicAttributeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateTopicAttributeRequest request = new UpdateTopicAttributeRequest();
        request.withTopicUrn("{topic_urn}");
        request.withName("{name}");
        UpdateTopicAttributeRequestBody body = new UpdateTopicAttributeRequestBody();
        body.withValue("{\"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\":\"_org_path_pub_0\",\"Effect\":\"Allow\",\"Principal\":{\"OrgPath\":{\"o-
bf966fe82ebb4d35d68b791729228788/r-001ebf32880a13eabfc8e1c37eee3ae9/
ou-0dbfffe92fd92ddb35feff9b4079459c\"}},\"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}\"}]}");
        request.withBody(body);
        try {
            UpdateTopicAttributeResponse response = client.updateTopicAttribute(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

```
}  
}  
}
```

- 更新主题策略

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
public class UpdateTopicAttributeSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before  
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
        // environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        SmnClient client = SmnClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateTopicAttributeRequest request = new UpdateTopicAttributeRequest();  
        request.withTopicUrn("{topic_urn}");  
        request.withName("{name}");  
        UpdateTopicAttributeRequestBody body = new UpdateTopicAttributeRequestBody();  
        body.withValue("");  
        request.withBody(body);  
        try {  
            UpdateTopicAttributeResponse response = client.updateTopicAttribute(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

- 更新主题策略

```
# coding: utf-8  
  
import os
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateTopicAttributeRequest()
        request.topic_urn = "{topic_urn}"
        request.name = "{name}"
        request.body = UpdateTopicAttributeRequestBody(
            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":"__org_path_pub_0","Effect":"Allow","Principal":{"OrgPath":["o-
                bf966fe82ebb4d35d68b791729228788/r-001ebf32880a13eabfc8e1c37eee3ae9/
                ou-0dbfffe92fd92ddb35feff9b4079459c"]},"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"}}]}
            )
        response = client.update_topic_attribute(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- 更新主题策略

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(SmnRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = UpdateTopicAttributeRequest()
    request.topic_urn = "{topic_urn}"
    request.name = "{name}"
    request.body = UpdateTopicAttributeRequestBody(
        value=""
    )
    response = client.update_topic_attribute(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- 更新主题策略

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateTopicAttributeRequest{
        request.TopicUrn = "{topic_urn}"
        request.Name = "{name}"
        request.Body = &model.UpdateTopicAttributeRequestBody{
            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\":\"__org_path_pub_0\",\"Effect\":\"Allow\",\"Principal\":{\"OrgPath\":[\"o-
                bf966fe82ebb4d35d68b791729228788/r-001ebf32880a13eabfc8e1c37eee3ae9/
                ou-0dbfffe92fd92ddb35feff9b4079459c\"]},\"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}\"}]}",
        }
    }
    response, err := client.UpdateTopicAttribute(request)
```

```
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 更新主题策略

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateTopicAttributeRequest{}
    request.TopicUrn = "{topic_urn}"
    request.Name = "{name}"
    request.Body = &model.UpdateTopicAttributeRequestBody{
        Value: "",
    }
    response, err := client.UpdateTopicAttribute(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.1.9 删除指定名称的主题策略

功能介绍

删除指定名称的主题策略。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-64 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
name	是	String	主题策略名称。只支持特定的策略名称，请参见 Topic属性表 。

请求参数

表 5-65 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-66 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-67 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-68 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-69 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-70 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除指定名称的主题策略

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

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.2 订阅操作

5.2.1 查询订阅者列表

功能介绍

分页返回请求者的所有的订阅列表，订阅列表按照订阅创建时间进行升序排列。分页查询可以指定offset以及limit。如果订阅者不存在，返回空列表。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/notifications/subscriptions

表 5-71 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

表 5-72 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量。 偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。
limit	否	Integer	查询数量限制。 取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。
protocol	否	String	协议名称，枚举值：http、https、sms、email、functionstage。

参数	是否必选	参数类型	描述
status	否	Integer	状态。 0: 未确认 1: 已确认 2: 不需要确认 3: 已取消确认 4: 已经删除。
endpoint	否	String	订阅终端。
fuzzy_remark	否	String	检索的订阅备注字段，模糊匹配。最大长度限制为128个字节。

请求参数

表 5-73 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-74 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
subscription_count	Integer	订阅者个数。
subscriptions	Array of ListSubscriptionsItem objects	Subscription结构体。

表 5-75 ListSubscriptionsItem

参数	参数类型	描述
topic_urn	String	Topic的唯一资源标识。
protocol	String	不同协议对应不同的endpoint（接受消息的接入点）。目前支持的协议包括： "default": 默认协议。 “email”：邮件传输协议，endpoint为邮箱地址。 “sms”：短信传输协议，endpoint为手机号码。 “functionstage”：FunctionGraph（函数）传输协议，endpoint为一个函数。 “http”、“https”：HTTP/HTTPS传输协议，endpoint为URL。
subscription_urn	String	订阅者的唯一资源标识。
owner	String	Topic创建者的项目ID。
endpoint	String	接受消息的接入点。
remark	String	备注。
status	Integer	订阅者状态：0表示订阅还未确认，1表示已经确认，3表示已经取消确认。
filter_policies	Array of SubscriptionsFilterPolicy objects	过滤策略。

表 5-76 SubscriptionsFilterPolicy

参数	参数类型	描述
name	String	过滤策略名称。 包含小写英文字母([a-z])、数字([0-9])、下划线(_)，下划线不得开始、结尾或连续出现)，长度限制[1,32]，不能是smn_开头。
string_equals	Array of strings	字符串精确匹配数组。数组长度[1, 10]，数组内容不能重复，值不能为null或者空字符串“”，长度限制[1,32]，中英文、数字、下划线

状态码：400

表 5-77 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-78 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-79 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-80 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询订阅者列表

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

响应示例

状态码：200

OK

```
{  
  "request_id" : "6a63a18b8bab40ffb71ebd9cb80d0085",  
  "subscription_count" : 2,  
  "subscriptions" : [ {  
    "topic_urn" : "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",  
    "protocol" : "sms",  
    "subscription_urn" :  
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837"  
  },  
    {  
      "owner" : "762bdb3251034f268af0e395c53ea09b",  
      "endpoint" : "xxxxxxxxxx",  
      "remark" : "",  
      "status" : 0,  
      "filter_policies" : [ {  
        "name" : "alarm",  
        "string_equals" : [ "os", "process" ]  
      }, {  
        "name" : "service",  
        "string_equals" : [ "api", "db" ]  
      } ]  
    }, {  
      "topic_urn" : "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",  
      "protocol" : "email",  
      "subscription_urn" :  
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:a2d52a9f5c3b47f48c3fafb177a58796",  
      "owner" : "762bdb3251034f268af0e395c53ea09b",  
      "endpoint" : "xx@example.com",  
      "remark" : "",  
      "status" : 0,  
      "filter_policies" : [ {  
        "name" : "alarm",  
        "string_equals" : [ "os", "process" ]  
      }, {  
        "name" : "service",  
        "string_equals" : [ "api", "db" ]  
      } ]  
    } ]  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.2.2 查询指定 Topic 的订阅者列表

功能介绍

分页获取特定Topic的订阅列表，订阅列表按照订阅创建时间进行升序排列。分页查询可以指定offset以及limit。如果指定Topic不存在订阅者，返回空列表。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-81 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

表 5-82 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量。 偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。
limit	否	Integer	查询的数量限制。 取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。
fuzzy_remark	否	String	检索的订阅备注字段，模糊匹配。最大长度限制为128字节。

请求参数

表 5-83 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-84 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
subscription_count	Integer	订阅者个数。
subscriptions	Array of ListSubscriptionsItem objects	Subscription结构体。

表 5-85 ListSubscriptionsItem

参数	参数类型	描述
topic_urn	String	Topic的唯一资源标识。
protocol	String	不同协议对应不同的endpoint（接受消息的接入点）。目前支持的协议包括： "default": 默认协议。 "email": 邮件传输协议，endpoint为邮箱地址。 "sms": 短信传输协议，endpoint为手机号码。 "functionstage": FunctionGraph（函数）传输协议，endpoint为一个函数。 "http"、"https": HTTP/HTTPS传输协议，endpoint为URL。
subscription_urn	String	订阅者的唯一资源标识。

参数	参数类型	描述
owner	String	Topic创建者的项目ID。
endpoint	String	接受消息的接入点。
remark	String	备注。
status	Integer	订阅者状态：0表示订阅还未确认，1表示已经确认，3表示已经取消确认。
filter_policies	Array of SubscriptionsFilterPolicy objects	过滤策略。

表 5-86 SubscriptionsFilterPolicy

参数	参数类型	描述
name	String	过滤策略名称。 包含小写英文字母([a-z])、数字([0-9])、下划线(_)，下划线不得开始、结尾或连续出现)，长度限制{1,32}，不能是smn_开头。
string_equals	Array of strings	字符串精确匹配数组。数组长度[1, 10]，数组内容不能重复，值不能为null或者空字符串“ ”，长度限制[1,32]，中英文、数字、下划线

状态码：400

表 5-87 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-88 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-89 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-90 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询名称为test_topic_v2的主题的订阅者列表

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

响应示例

状态码：200

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "subscription_count": 2,
  "subscriptions": [ {
    "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
    "protocol": "sms",
    "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837"
```

```

"owner": "762bdb3251034f268af0e395c53ea09b",
"endpoint": "xxxxxxxx",
"remark": "",
"status": 0,
"filter_policies": [ {
  "name": "alarm",
  "string_equals": [ "os", "process" ]
}, {
  "name": "service",
  "string_equals": [ "api", "db" ]
} ]
}, {
  "topic_urn": "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1",
  "protocol": "email",
  "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:a2d52a9f5c3b47f48c3fafb177a58796",
  "owner": "762bdb3251034f268af0e395c53ea09b",
  "endpoint": "xx@example.com",
  "remark": "",
  "status": 0,
  "filter_policies": [ {
    "name": "alarm",
    "string_equals": [ "os", "process" ]
  }, {
    "name": "service",
    "string_equals": [ "api", "db" ]
  } ]
} ]
} ]
}

```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.2.3 订阅

功能介绍

为指定Topic添加一个订阅者，如果订阅者的状态为未确认，则向订阅者发送一个确认的消息。待订阅者进行ConfirmSubscription确认后，该订阅者才能收到Topic发布的消息。单Topic默认可添加10000个订阅者，高并发场景下，可能会出现订阅者数量超过10000仍添加成功的情况，此为正常现象。接口是幂等的，如果添加已存在的订阅者，则返回成功，且status code为200，否则status code为201。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-91 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-92 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-93 请求 Body 参数

参数	是否必选	参数类型	描述
protocol	是	String	<p>不同协议对应不同的endpoint（接受消息的接入点）。</p> <p>目前支持的协议包括：</p> <p>“email”：邮件传输协议，endpoint为邮箱地址。</p> <p>“sms”：短信传输协议，endpoint为手机号码。</p> <p>“functionstage”：FunctionGraph（函数）传输协议，endpoint为一个函数。</p> <p>“functiongraph”：FunctionGraph（工作流）传输协议，endpoint为由一组函数编排成的工作流。</p> <p>“http”、“https”：HTTP/HTTPS传输协议，endpoint为URL。</p> <p>“callnotify”：语音通知传输协议，endpoint为手机号码。</p> <p>“wechat”：微信群机器人传输协议。</p> <p>“dingding”：钉钉群机器人传输协议。</p> <p>“feishu”：飞书群机器人传输协议。</p> <p>“welink”：welink群机器人传输协议。</p> <p>“dingTalkBot”：个人钉钉传输协议。</p>

参数	是否必选	参数类型	描述
endpoint	是	String	<p>说明：</p> <p>http协议，接入点必须以“http://”开头。</p> <p>https协议，接入点必须以“https://”开头。</p> <p>email协议，接入点必须是邮件地址。</p> <p>sms协议，接入点必须是一个电话号码。</p> <p>functionstage协议，接入点必须是一个函数。</p> <p>functiongraph协议，接入点必须是一个函数工作流。</p> <p>callnotify协议，接入点必须是一个电话号码。</p> <p>dingding协议，接入点必须是一个钉钉自定义群机器人的地址，或添加了钉钉企业内部机器人的群组openConversationId。添加钉钉企业内部机器人对应的群组openConversationId时，该字段不能以“http://”或“https://”为开头。</p> <p>wechat协议，接入点必须是一个微信群机器人的地址。</p> <p>feishu协议，接入点必须是一个飞书群机器人的地址。</p> <p>welink协议，接入点必须是一个welink的群号。</p> <p>dingTalkBot协议，接入点必须是一个钉钉企业用户的userId。</p>
remark	否	String	备注。最大支持128字节，约42个中文，必须是UTF-8编码的字符串，否则无法正常显示中文。
extension	否	SubscriptionExtension object	扩展信息。
subscriptions	否	Array of BatchAddSubscriptionsRequest objects	当需要批量创建订阅时，需要传入该字段。SMN支持一次最多创建50个订阅。

表 5-94 BatchAddSubscriptionsRequestBody

参数	是否必选	参数类型	描述
protocol	是	String	不同协议对应不同的endpoint（接受消息的接入点）。 目前支持的协议包括： “email”：邮件传输协议，endpoint为邮箱地址。 “sms”：短信传输协议，endpoint为手机号码。 “functionstage”：FunctionGraph（函数）传输协议，endpoint为一个函数。 “functiongraph”：FunctionGraph（工作流）传输协议，endpoint为由一组函数编排成的工作流。 “http”、“https”：HTTP/HTTPS传输协议，endpoint为URL。 “callnotify”：语音通知传输协议，endpoint为手机号码。 “wechat”：微信群机器人传输协议。 “dingding”：钉钉群机器人传输协议。 “feishu”：飞书群机器人传输协议。 “welink”：welink群机器人传输协议。

参数	是否必选	参数类型	描述
endpoint	是	String	说明： http协议，接入点必须以“http://”开头。 https协议，接入点必须以“https://”开头。 email协议，接入点必须是邮件地址。 sms协议，接入点必须是一个电话号码。 functionstage协议，接入点必须是一个函数。 functiongraph协议，接入点必须是一个函数工作流。 callnotify协议，接入点必须是一个电话号码。 dingding协议，接入点必须是一个钉钉自定义群机器人的地址，或添加了钉钉企业内部机器人的群组openConversationId。添加钉钉企业内部机器人对应的群组openConversationId时，该字段不能以“http://”或“https://”为开头。 wechat协议，接入点必须是一个微信群机器人的地址。 feishu协议，接入点必须是一个飞书群机器人的地址。 welink协议，接入点必须是一个welink的群号。
remark	否	String	备注。最大支持128字节，约42个中文，必须是UTF-8编码的字符串，否则无法正常显示中文。
extension	否	SubscriptionExtension object	订阅的扩展信息。

表 5-95 SubscriptionExtension

参数	是否必选	参数类型	描述
client_id	否	String	这是应用ID字段。当protocol值为welink时，该字段为必填字段，从welink方获取。

参数	是否必选	参数类型	描述
client_secret	否	String	该字段为应用secret字段。当protocol值为welink时，该字段为必填字段，从welink方获取。
keyword	否	String	该字段为关键字字段。当protocol值为feishu时，这里的keyword字段和sign_secret字段二者必选其一。当用户在飞书或钉钉自定义机器人端添加关键字校验的安全策略时，这里的关键字必须是飞书或钉钉自定义机器人中所填写的关键字之一。
sign_secret	否	String	这是加签密钥字段。当protocol协议为feishu时，这个字段和keyword字段二者必选且只能选其一。密钥配置必须与客户在飞书或钉钉自定义机器人的密钥配置完全一致。例如，如果在飞书端配置了密钥并且没有配置关键字，则在此处填入从飞书获取的密钥字段，如果在飞书端没有配置密钥并且配置了关键字，则不填写该字段。
header	否	Map<String,String>	<p>该字段为http header字段，用户可以在字段限制范围内自定义http header，header字段内容以KV对形式存在。当使用主题发送时，已确认的订阅发送消息会携带用户自定义的http header。</p> <p>header应满足如下要求：</p> <p>key值限定为：包含英文字母([A-Za-z])、数字([0-9])、中划线(-)hyphens，中划线不得作为结尾，不得连续出现。</p> <p>K/V不得超过10个</p> <p>key需要以"x-"开头，不能以"x-smn"开头，正确示例：x-abc-cba, x-abc</p> <p>所有K/V长度总和不得超过1024个字符</p> <p>key不区分大小写</p> <p>key值不可重复</p> <p>value值限定为ASCII码，不支持中文或其他Unicode，支持空格</p>

参数	是否必选	参数类型	描述
app_key	否	String	个人钉钉appKey字段，字符长度限制64个，仅支持字母、数字、中划线(-)、下划线(_)。当订阅协议为dingTalkBot时，该字段必选。
app_secret	否	String	个人钉钉appSecret字段，字符长度限制128个，仅支持字母、数字、中划线(-)、下划线(_)。当订阅协议为dingTalkBot时，该字段必选。
robot_code	否	String	个人钉钉robotCode字段，名称：机器人编码，字符长度限制64个，仅支持字母、数字、中划线(-)、下划线(_)，一般与appKey一致。当订阅协议为dingTalkBot时，该字段必选。

响应参数

状态码：201

表 5-96 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
subscription_urn	String	订阅者的唯一资源标识。

状态码：400

表 5-97 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-98 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-99 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-100 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

在名称为test_topic_v1的主题下添加一个邮件订阅者

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1/subscriptions
```

```
{  
  "protocol": "email",  
  "endpoint": "xxx@example.com",  
  "remark": "运维"  
}
```

响应示例

状态码：201

OK

```
[ {
  "request_id" : "6a63a18b8bab40ffb71ebd9cb80d0085",
  "subscription_urn" :
  "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837
"
} ]
```

SDK 代码示例

SDK代码示例如下。

Java

在名称为test_topic_v1的主题下添加一个邮件订阅者

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class AddSubscriptionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        AddSubscriptionRequest request = new AddSubscriptionRequest();
        request.withTopicUrn("{topic_urn}");
        AddSubscriptionRequestBody body = new AddSubscriptionRequestBody();
        body.withRemark("运维");
        body.withEndpoint("xxx@example.com");
        body.withProtocol("email");
        request.withBody(body);
        try {
            AddSubscriptionResponse response = client.addSubscription(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
        }
    }
}
```

```
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

在名称为test_topic_v1的主题下添加一个邮件订阅者

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddSubscriptionRequest()
        request.topic_urn = "{topic_urn}"
        request.body = AddSubscriptionRequestBody(
            remark="运维",
            endpoint="xxx@example.com",
            protocol="email"
        )
        response = client.add_subscription(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

在名称为test_topic_v1的主题下添加一个邮件订阅者

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.AddSubscriptionRequest{}
request.TopicUrn = "{topic_urn}"
remarkAddSubscriptionRequestBody := "运维"
request.Body = &model.AddSubscriptionRequestBody{
    Remark: &remarkAddSubscriptionRequestBody,
    Endpoint: "xxx@example.com",
    Protocol: "email",
}
response, err := client.AddSubscription(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.2.4 取消订阅

功能介绍

删除指定的订阅者。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-101 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
subscription_urn	是	String	订阅者的唯一的资源标识，可通过 查询订阅者列表 获取该标识。

请求参数

表 5-102 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-103 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-104 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-105 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-106 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-107 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除订阅URN为
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837的订阅者

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

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.2.5 更新订阅者

功能介绍

更新订阅者备注。

调用方法

请参见[如何调用API](#)。

URI

```
PUT /v2/{project_id}/notifications/topics/{topic_urn}/subscriptions/  
{subscription_urn}
```


表 5-108 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识。可以通过 查看主题列表 获取该标识。
subscription_urn	是	String	订阅者的唯一的资源标识，可通过 查询订阅者列表 获取该标识。

请求参数

表 5-109 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-110 请求 Body 参数

参数	是否必选	参数类型	描述
remark	是	String	订阅者备注。订阅者备注的最大长度为128byte。

响应参数

状态码：200

表 5-111 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
subscription_urn	String	订阅者的唯一资源标识。

状态码：400

表 5-112 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-113 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-114 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-115 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新订阅者

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1/subscriptions/  
urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837  
  
{  
  "remark" : "test remark"  
}
```

响应示例

状态码：200

OK

```
{  
  "request_id" : "fdbabe38ead6482b8574f82a3d1168e9",  
  "subscription_urn" :  
  "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新订阅者

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
public class UpdateSubscriptionSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        SmnClient client = SmnClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateSubscriptionRequest request = new UpdateSubscriptionRequest();  
        request.withTopicUrn("{topic_urn}");  
        request.withSubscriptionUrn("{subscription_urn}");  
        UpdateSubscriptionRequestBody body = new UpdateSubscriptionRequestBody();
```

```
body.withRemark("test remark");
request.withBody(body);
try {
    UpdateSubscriptionResponse response = client.updateSubscription(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

更新订阅者

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateSubscriptionRequest()
        request.topic_urn = "{topic_urn}"
        request.subscription_urn = "{subscription_urn}"
        request.body = UpdateSubscriptionRequestBody(
            remark="test remark"
        )
        response = client.update_subscription(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新订阅者

```
package main
```

```
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := smn.NewSmnClient(  
        smn.SmnClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.UpdateSubscriptionRequest{}  
    request.TopicUrn = "{topic_urn}"  
    request.SubscriptionUrn = "{subscription_urn}"  
    request.Body = &model.UpdateSubscriptionRequestBody{  
        Remark: "test remark",  
    }  
    response, err := client.UpdateSubscription(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.3 模板操作

5.3.1 查询消息模板列表

功能介绍

分页查询模板列表，模板列表按照创建时间进行升序排列。分页查询可以指定offset以及limit。如果不存在模板，则返回空列表。额外的查询参数分别有message_template_name和protocol。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/notifications/message_template

表 5-116 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

表 5-117 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量，偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。
limit	否	Integer	取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。
message_template_name	否	String	模板的名称。 只能包含大写字母、小写字母、数字、-和_，且必须由大写字母、小写字母或数字开头，长度在1到64个字符之间。

参数	是否必选	参数类型	描述
protocol	否	String	模板支持的协议类型。 目前支持的协议包括： “default”：默认协议。 “email”：邮件传输协议。 “sms”：短信传输协议。 “functionstage”： FunctionGraph（函数）传输协议。 “http”、“https”：HTTP/ HTTPS传输协议。

请求参数

表 5-118 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-119 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
message_template_count	Integer	返回的模板个数。
message_templates	Array of MessageTemplate objects	Message_template结构体数组。

表 5-120 MessageTemplate

参数	参数类型	描述
message_template_id	String	模板ID。
message_template_name	String	模板名称。
protocol	String	模板协议类型。 目前支持的协议包括： “email”：邮件传输协议。 “sms”：短信传输协议。 “functionstage”：FunctionGraph（函数）传输协议。 “http”、“https”：HTTP/HTTPS传输协议。
tag_names	Array of strings	模板tag列表
create_time	String	模板创建时间 时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
update_time	String	模板最后更新时间 时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。

状态码：400

表 5-121 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-122 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-123 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-124 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询消息模板列表

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

响应示例

状态码：200

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "message_template_count": 3,
  "message_templates": [ {
    "message_template_name": "confirm_message",
    "protocol": "sms",
    "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" : "57ba8dcecca844878c5dd5815b65d10f"  
}]  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.3.2 创建消息模板

功能介绍

创建一个模板，用户可以按照模板去发送消息，这样可以减少请求的数据量。

单用户默认可创建100个消息模板，高并发场景下，可能会出现消息模板数量超过100仍创建成功的情况，此为正常现象。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/notifications/message_template

表 5-125 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

请求参数

表 5-126 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-127 请求 Body 参数

参数	是否必选	参数类型	描述
message_template_name	是	String	创建模板的名称。只能包含大写字母、小写字母、数字、-和_，且必须由大写字母、小写字母或数字开头，长度在1到64个字符之间。
protocol	是	String	模板支持的协议类型。 目前支持的协议包括： “default”：默认协议。 “email”：邮件传输协议。 “sms”：短信传输协议。 “functionstage”：FunctionGraph（函数）传输协议。 “http”、“https”：HTTP/HTTPS传输协议。
content	是	String	模板内容，模板目前仅支持纯文本模式。模板内容不能空，最大支持256KB。

响应参数

状态码：201

表 5-128 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
message_template_id	String	模板唯一的资源标识。

状态码：400

表 5-129 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-130 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-131 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-132 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

创建模板名称为confirm_message，且模板支持的协议类型为https的消息模板

```
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.)"
}
```

响应示例

状态码：201

OK

```
{
  "request_id" : "6a63a18b8bab40ffb71ebd9cb80d0085",
  "message_template_id" : "57ba8dcecd844878c5dd5815b65d10f"
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建模板名称为confirm_message，且模板支持的协议类型为https的消息模板

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class CreateMessageTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateMessageTemplateRequest request = new CreateMessageTemplateRequest();
```

```
    CreateMessageTemplateRequestBody body = new CreateMessageTemplateRequestBody();
    body.withContent("(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.)");
    body.withProtocol("https");
    body.withMessageTemplateName("confirm_message");
    request.withBody(body);
    try {
        CreateMessageTemplateResponse response = client.createMessageTemplate(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

创建模板名称为confirm_message，且模板支持的协议类型为https的消息模板

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateMessageTemplateRequest()
        request.body = CreateMessageTemplateRequestBody(
            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.)",
            protocol="https",
            message_template_name="confirm_message"
        )
        response = client.create_message_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建模板名称为confirm_message，且模板支持的协议类型为https的消息模板

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateMessageTemplateRequest{}
    request.Body = &model.CreateMessageTemplateRequestBody{
        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.)",
        Protocol: "https",
        MessageTemplateName: "confirm_message",
    }
    response, err := client.CreateMessageTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized

状态码	描述
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.3.3 查询消息模板详情

功能介绍

查询模板详情，包括模板内容。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-133 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
message_template_id	是	String	模板唯一的资源标识，可通过查询 消息模板列表 获取该标识。

请求参数

表 5-134 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-135 响应 Body 参数

参数	参数类型	描述
message_template_id	String	模板ID。
message_template_name	String	模板名称。
protocol	String	模板支持的协议类型。 目前支持的协议包括： “default”：默认协议。 “email”：邮件传输协议。 “sms”：短信传输协议。 “functionstage”：FunctionGraph（函数）传输协议。 “http”、“https”：HTTP/HTTPS传输协议。
tag_names	Array of strings	模板tag列表。 是消息模板“{}”内的字段，在具体使用消息模板时，可根据实际情况替为该字段赋值。
create_time	String	模板创建时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
update_time	String	模板最后更新时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
content	String	模板内容。
request_id	String	请求的唯一标识ID。

状态码：400

表 5-136 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-137 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-138 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-139 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询消息模板详情

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template/  
57ba8dcecd844878c5dd5815b65d10f
```

```
null
```

响应示例

状态码：200

OK

```
{  
  "message_template_id": "57ba8dcecd844878c5dd5815b65d10f",  
  "message_template_name": "confirm_message",  
  "protocol": "https",
```

```
"tag_names" : [ "topic_id_id4" ],  
"create_time" : "2016-08-02T08:22:20Z",  
"update_time" : "2016-08-02T08:22:25Z",  
"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.)",  
"request_id" : "ba79ca8f794f4f50985ce7b98a401b47"  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.3.4 更新消息模板

功能介绍

修改消息模板的内容。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-140 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
message_template_id	是	String	模板唯一的资源标识，可通过查询 消息模板列表 获取该标识。

请求参数

表 5-141 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-142 请求 Body 参数

参数	是否必选	参数类型	描述
content	是	String	模板内容。

响应参数

状态码：200

表 5-143 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-144 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-145 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-146 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-147 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新消息模板内容

```
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.)"
}
```

响应示例

状态码：200

OK

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

SDK 代码示例

SDK代码示例如下。

Java

更新消息模板内容

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class UpdateMessageTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateMessageTemplateRequest request = new UpdateMessageTemplateRequest();
        request.withMessageTemplateId("{message_template_id}");
        UpdateMessageTemplateRequestBody body = new UpdateMessageTemplateRequestBody();
        body.withContent("(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.)");
        request.withBody(body);
        try {
            UpdateMessageTemplateResponse response = client.updateMessageTemplate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

更新消息模板内容

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateMessageTemplateRequest()
        request.message_template_id = "{message_template_id}"
        request.body = UpdateMessageTemplateRequestBody(
            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 = client.update_message_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

更新消息模板内容

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
```

```
client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateMessageTemplateRequest{
    request.MessageTemplateId = "{message_template_id}"
    request.Body = &model.UpdateMessageTemplateRequestBody{
        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, err := client.UpdateMessageTemplate(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.3.5 删除消息模板

功能介绍

删除消息模板。删除模板之前的消息请求都可以使用该模板发送，删除之后无法再使用该模板发送消息。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-148 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
message_template_id	是	String	模板唯一的资源标识，可通过查询 消息模板列表 获取该标识。

请求参数

表 5-149 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-150 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码：400

表 5-151 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-152 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-153 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-154 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除消息模板

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

```
null
```

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.4 发布消息操作

5.4.1 消息发布

功能介绍

将消息发送给Topic的所有订阅端点。当返回消息ID时，该消息已被保存并开始尝试将其推送到Topic的订阅者。为确保您的消息能够成功推送到各个订阅者，请确保您的消息内容符合当地法律法规要求。

三种消息发送方式

message

message_structure

message_template_name

只需要设置其中一个，如果同时设置，生效的优先级为

message_structure > message_template_name > message。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-155 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-156 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-157 请求 Body 参数

参数	是否必选	参数类型	描述
subject	否	String	消息标题，给邮箱订阅者发送邮件时作为邮件主题，长度不能超过512个字节。
message	否	String	发送的消息。消息体必须是UTF-8编码的字符串，大小至多256KB。如果订阅者是手机号码，短信长度限制为490字，超出则可能被运营商拦截。短信内容不能包含“[]”或者“【】”符号。说明：三种消息发送方式：message、message_structure、message_template_name至少设置其中一个，如果同时设置，生效的优先级为message_structure > message_template_name > message。

参数	是否必选	参数类型	描述
message_structure	否	String	<p>Json格式的字符串。支持“email”、“sms”、“http”、“https”、“functiongraph”、“functionstage”、“dingding”、“wechat”、“feishu”、“welink”。必须设置默认的消息“default”，当匹配不到消息协议时，按“default”中的内容发送。</p> <ul style="list-style-type: none"> - 其中，钉钉、微信、飞书、welink协议类型的消息需指定msgType字段；钉钉，微信和飞书机器人协议支持msgType为text（纯文本）和markdown（MD）格式消息，welink机器人类型暂仅支持msgType为text的纯文本消息。 - 钉钉机器人协议支持通过at字段实现@群组成员。当您需要@群成员时，可以在isAtAll字段中传入布尔值，表示是否需要@群组内所有人。您可以在atMobiles字段中传入需要@的人的手机号列表，或在atUserIds字段中传入需要@的钉钉userID列表。当您使用atMobiles字段或atUserIds字段时，需要在消息内容中同时传入@对应手机号或userID的信息。展示效果请参考钉钉官方文档。 - 飞书机器人协议的markdown消息支持通过传入color字段设置标题的底色。能够支持的color字段可参考飞书官方文档。 <p>说明：三种消息发送方式：message、message_structure、message_template_name至少设置其中一个，如果同时设置，生效的优先级为message_structure > message_template_name > message。</p>

参数	是否必选	参数类型	描述
message_template_name	否	String	消息模板名称，可通过 查询消息模板列表 获取名称。说明：三种消息发送方式：message、message_structure、message_template_name至少设置其中一个，如果同时设置，生效的优先级为 message_structure > message_template_name > message。
tags	否	Map<String,String>	tag以及替换tag的参数组成的字典。消息模板中的标签对应的值。使用消息模板方式的消息发布必须携带该参数。字典中的key为消息模板中的参数名称，不超过21个字符。字典中的value为消息模板中的参数被替换后的值，不超过1KB。
time_to_live	否	String	指消息在SMN系统内部的最长存留时间。超过该存留时间，系统将不再发送该消息。单位是s，变量默认值是3600s，即一小时。值为正整数且小于等于3600*24。
message_attributes	否	Array of MessageAttribute objects	消息属性列表
locale	否	String	语言，发送出去的消息中SMN附加系统内容的语言，若没传入，使用账号的语言。取值范围是该局点支持的语言，比如：zh-cn,en-us等

表 5-158 MessageAttribute

参数	是否必选	参数类型	描述
name	是	String	属性名称。属性名称只能包含小写英文字母([a-z])、数字([0-9])、下划线(_)，下划线不能出现在开始或结尾，下划线不能连续出现，长度为1到32个字符

参数	是否必选	参数类型	描述
type	是	String	属性类型 STRING: 字符串 (String) 类型 STRING_ARRAY: 字符串数组 (String.Array) 类型 PROTOCOL: 协议类型
value	是	Object	属性值。 当属性类型为“STRING”时, 属性值只能包含中英文、数字、下划线, 长度为1到32个字符。 当属性类型为“STRING_ARRAY”时, 属性值为字符串数组, 数组长度为1到10, 数组中的元素内容不能重复, 数组中的每个字符串都只能包含中英文、数字、下划线, 且长度为1到32个字符。 当属性类型为“PROTOCOL”时, 属性值为支持协议类型的字符串数组。

响应参数

状态码: 200

表 5-159 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
message_id	String	唯一的消息ID。

状态码: 400

表 5-160 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403**表 5-161 响应 Body 参数**

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404**表 5-162 响应 Body 参数**

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：429**表 5-163 响应 Body 参数**

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500**表 5-164 响应 Body 参数**

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

● 消息发布

```
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",  
  "message_attributes": [ {  
    "name": "smn_protocol",  
    "type": "PROTOCOL",  
    "value": [ "email", "sms" ]  
  } ]  
}
```

● 使用消息结构体方式的消息发布

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

```
{  
  "subject": "test message v2",  
  "message_structure": "{ \"default\": \"Message structure test message v2\", \"sms\": \"Message structure test message v2\", \"email\": \"Message structure test message v2\", \"wechat\": \"{\\\"msgtype\\\":\\\"text\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\", \"dingding\": \"{\\\"msgtype\\\":\\\"markdown\\\",\\\"at\\\":{\\\"atMobiles\\\": [\\\"15XXXXXX\\\"],\\\"atUserIds\\\": [\\\"1234567XXX\\\"]},\\\"isAtAll\\\": true},\\\"markdown\\\":{\\\"content\\\":\\\"Message structure test message v2 @15XXXXXX @1234567XXX\\\",\\\"title\\\":\\\"Message structure title\\\"}}\", \"feishu\": \"{\\\"msgtype\\\":\\\"text\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\", \"welinkRed\": \"{\\\"msgtype\\\":\\\"text\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\", \"http\": \"Message structure test message v2\", \"https\": \"Message structure test message v2\", \"dingTalkBot\": \"{\\\"msgtype\\\":\\\"markdown\\\",\\\"markdown\\\":{\\\"content\\\":\\\"Message structure test message v2 dingTalkBot markdown\\\",\\\"title\\\":\\\"Message structure title\\\"}}\", \"functionstage\": \"Message structure test message v2\" }",  
  "time_to_live": "3600",  
  "message_attributes": [ {  
    "name": "smn_protocol",  
    "type": "PROTOCOL",  
    "value": [ "email", "sms" ]  
  } ]  
}
```

● 使用消息模板方式的消息发布

```
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"  
  },  
  "message_attributes": [ {  
    "name": "smn_protocol",  
    "type": "PROTOCOL",  
    "value": [ "email", "sms" ]  
  } ]  
}
```

响应示例

状态码: 200

OK

```
{
  "request_id" : "6a63a18b8bab40ffb71ebd9cb80d0085",
  "message_id" : "bf94b63a5dfb475994d3ac34664e24f2"
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 消息发布

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class PublishMessageSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        PublishMessageRequest request = new PublishMessageRequest();
        request.withTopicUrn("{topic_urn}");
        PublishMessageRequestBody body = new PublishMessageRequestBody();
        List<MessageAttribute> listbodyMessageAttributes = new ArrayList<>();
        listbodyMessageAttributes.add(
            new MessageAttribute()
                .withName("smn_protocol")
                .withType(MessageAttribute.TypeEnum.fromValue("PROTOCOL"))
                .withValue("[email, sms]"));
        body.withMessageAttributes(listbodyMessageAttributes);
        body.withTimeToLive("3600");
        body.withMessage("Message test message v2");
        body.withSubject("test message v2");
        request.withBody(body);
        try {
            PublishMessageResponse response = client.publishMessage(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 使用消息结构体方式的消息发布

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class PublishMessageSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        PublishMessageRequest request = new PublishMessageRequest();
        request.withTopicUrn("{topic_urn}");
        PublishMessageRequestBody body = new PublishMessageRequestBody();
        List<MessageAttribute> listbodyMessageAttributes = new ArrayList<>();
        listbodyMessageAttributes.add(
            new MessageAttribute()
                .withName("smn_protocol")
                .withType(MessageAttribute.TypeEnum.fromValue("PROTOCOL"))
                .withValue("[email, sms]"));
        body.withMessageAttributes(listbodyMessageAttributes);
        body.withTimeToLive("3600");
        body.withMessageStructure("{\" default\": \"Message structure test message v2\", \"sms\": \"Message structure test message v2\", \"email\": \"Message structure test message v2\", \"wechat\": \"Message structure test message v2\", \"dingding\": \"Message structure test message v2\", \"feishu\": \"Message structure test message v2\", \"http\": \"Message structure test message v2\", \"markdown\": \"Message structure test message v2 @15XXXXXX\", \"isAtAll\": true, \"atMobiles\": [\"15XXXXXX\"], \"atUserIds\": [\"1234567XXX\"], \"content\": \"Message structure test message v2\", \"title\": \"Message structure title\"}");
    }
}
```



```
);  
Map<String, String> listbodyTags = new HashMap<>();  
listbodyTags.put("topic_urn", "topic_urn3331");  
listbodyTags.put("topic_id", "topic_id3332");  
body.withMessageAttributes(listbodyMessageAttributes);  
body.withTimeToLive("3600");  
body.withTags(listbodyTags);  
body.withMessageTemplateName("confirm_message");  
body.withSubject("test message template v2");  
request.withBody(body);  
try {  
    PublishMessageResponse response = client.publishMessage(request);  
    System.out.println(response.toString());  
} catch (ConnectionException e) {  
    e.printStackTrace();  
} catch (RequestTimeoutException e) {  
    e.printStackTrace();  
} catch (ServiceResponseException e) {  
    e.printStackTrace();  
    System.out.println(e.getHttpStatusCode());  
    System.out.println(e.getRequestId());  
    System.out.println(e.getErrorCode());  
    System.out.println(e.getErrorMsg());  
}  
}
```

Python

- 消息发布

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdksmn.v2 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    # environment variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before  
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    # environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = SmnClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = PublishMessageRequest()  
        request.topic_urn = "{topic_urn}"  
        listMessageAttributesbody = [  
            MessageAttribute(  
                name="smn_protocol",  
                type="PROTOCOL",  
                value="[email, sms]"  
            )  
        ]  
        request.body = PublishMessageRequestBody(  
            message_attributes=listMessageAttributesbody,  
            time_to_live="3600",
```

```
        message="Message test message v2",
        subject="test message v2"
    )
    response = client.publish_message(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 使用消息结构体方式的消息发布

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = PublishMessageRequest()
        request.topic_urn = "{topic_urn}"
        listMessageAttributesbody = [
            MessageAttribute(
                name="smn_protocol",
                type="PROTOCOL",
                value="[email, sms]"
            )
        ]
        request.body = PublishMessageRequestBody(
            message_attributes=listMessageAttributesbody,
            time_to_live="3600",
            message_structure="{ 'default': 'Message structure test message v2', 'sms': 'Message
            structure test message v2', 'email': 'Message structure test message v2', 'wechat': '{\msgtype
            \":\text\","text\":"content\":"Message structure test message v2\","dingding": "{\msgtype
            \":\markdown\","at\":"atMobsiles\":"[\"15XXXXXX\"],"atUserIds\":"[\"1234567XXX\"],"isAtAll\":"
            true\","markdown\":"content\":"Message structure test message v2 @15XXXXXX @1234567XXX
            \","title\":"Message structure title\","feishu": "{\msgtype\":"text\","text\":"content
            \":"Message structure test message v2\","welinkRed": "{\msgtype\":"text\","text\":"content
            \":"Message structure test message v2\","http": "Message structure test message v2", "https":
            "Message structure test message v2", "dingTalkBot": "{\msgtype\":"markdown\","markdown\":"
            {\content\":"Message structure test message v2 dingTalkBot markdown\","title\":"Message
            structure title\","functionstage": "Message structure test message v2 }",
            subject="test message v2"
        )
        response = client.publish_message(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

- 使用消息模板方式的消息发布

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = PublishMessageRequest()
        request.topic_urn = "{topic_urn}"
        listMessageAttributesbody = [
            MessageAttribute(
                name="smn_protocol",
                type="PROTOCOL",
                value="[email, sms]"
            )
        ]
        listTagsbody = {
            "topic_urn": "topic_urn3331",
            "topic_id": "topic_id3332"
        }
        request.body = PublishMessageRequestBody(
            message_attributes=listMessageAttributesbody,
            time_to_live="3600",
            tags=listTagsbody,
            message_template_name="confirm_message",
            subject="test message template v2"
        )
        response = client.publish_message(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 消息发布

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.PublishMessageRequest{}
    request.TopicUrn = "{topic_urn}"
    valueMessageAttributes := "[email, sms]"
    var valueMessageAttributesInterface interface{} = valueMessageAttributes
    var listMessageAttributesbody = []model.MessageAttribute{
        {
            Name: "smn_protocol",
            Type: model.GetMessageAttributeTypeEnum().PROTOCOL,
            Value: &valueMessageAttributesInterface,
        },
    }
    }
    timeToLivePublishMessageRequestBody := "3600"
    messagePublishMessageRequestBody := "Message test message v2"
    subjectPublishMessageRequestBody := "test message v2"
    request.Body = &model.PublishMessageRequestBody{
        MessageAttributes: &listMessageAttributesbody,
        TimeToLive: &timeToLivePublishMessageRequestBody,
        Message: &messagePublishMessageRequestBody,
        Subject: &subjectPublishMessageRequestBody,
    }
    response, err := client.PublishMessage(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
}
```

- 使用消息结构体方式的消息发布

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
```



```
// In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.PublishMessageRequest{}
request.TopicUrn = "{topic_urn}"
valueMessageAttributes := "[email, sms]"
var valueMessageAttributesInterface interface{} = valueMessageAttributes
var listMessageAttributesbody = []model.MessageAttribute{
    {
        Name: "smn_protocol",
        Type: model.GetMessageAttributeTypeEnum().PROTOCOL,
        Value: &valueMessageAttributesInterface,
    },
}
timeToLivePublishMessageRequestBody := "3600"
messageStructurePublishMessageRequestBody := "{ \"default\": \"Message structure test message
v2\", \"sms\": \"Message structure test message v2\", \"email\": \"Message structure test message v2\",
\"wechat\": \"{\\\"msgtype\\\":\\\"text\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\",
\"dingding\": \"{\\\"msgtype\\\":\\\"markdown\\\",\\\"at\\\":{\\\"atMobiles\\\": [\\\"15XXXXXX\\\"],\\\"atUserIds\\\":
[\\\"1234567XXX\\\"],\\\"isAtAll\\\": true},\\\"markdown\\\":{\\\"content\\\":\\\"Message structure test message v2
@15XXXXXX @1234567XXX\\\",\\\"title\\\":\\\"Message structure title\\\"}}\", \"feishu\": \"{\\\"msgtype\\\":\\\"text
\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\", \"welinkRed\": \"{\\\"msgtype\\\":\\\"text
\\\",\\\"text\\\":{\\\"content\\\":\\\"Message structure test message v2\\\"}}\", \"http\": \"Message structure test
message v2\", \"https\": \"Message structure test message v2\", \"dingTalkBot\": \"{\\\"msgtype
\\\":\\\"markdown\\\",\\\"markdown\\\":{\\\"content\\\":\\\"Message structure test message v2 dingTalkBot
markdown\\\",\\\"title\\\":\\\"Message structure title\\\"}}\", \"functionstage\": \"Message structure test
message v2\" }"
subjectPublishMessageRequestBody := "test message v2"
request.Body = &model.PublishMessageRequestBody{
    MessageAttributes: &listMessageAttributesbody,
    TimeToLive: &timeToLivePublishMessageRequestBody,
    MessageStructure: &messageStructurePublishMessageRequestBody,
    Subject: &subjectPublishMessageRequestBody,
}
response, err := client.PublishMessage(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 使用消息模板方式的消息发布

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.PublishMessageRequest{}
request.TopicUrn = "{topic_urn}"
valueMessageAttributes := "[email, sms]"
var valueMessageAttributesInterface interface{} = valueMessageAttributes
var listMessageAttributesbody = []model.MessageAttribute{
    {
        Name: "smn_protocol",
        Type: model.GetMessageAttributeTypeEnum().PROTOCOL,
        Value: &valueMessageAttributesInterface,
    },
}
var listTagsbody = map[string]string{
    "topic_urn": "topic_urn3331",
    "topic_id": "topic_id3332",
}
timeToLivePublishMessageRequestBody := "3600"
messageTemplateNamePublishMessageRequestBody := "confirm_message"
subjectPublishMessageRequestBody := "test message template v2"
request.Body = &model.PublishMessageRequestBody{
    MessageAttributes: &listMessageAttributesbody,
    TimeToLive: &timeToLivePublishMessageRequestBody,
    Tags: listTagsbody,
    MessageTemplateName: &messageTemplateNamePublishMessageRequestBody,
    Subject: &subjectPublishMessageRequestBody,
}
response, err := client.PublishMessage(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
403	Unauthorized
404	Not Found
429	Too Many Requests
500	Internal Server Error

错误码

请参见[错误码](#)。

5.4.2 发布探测消息

功能介绍

基于主题发送http/https探测消息，探测当前http/https 终端是否可用，SMN出口是否能够正常访问该终端。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-165 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-166 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-167 请求 Body 参数

参数	是否必选	参数类型	描述
protocol	是	String	协议类型，当前仅支持http/https，不可为空
endpoint	是	String	待探测的终端地址，当前仅支持以http:// 或https://开头，不可为空
extension	否	extension object	Http/Https协议可以指定多个K/V对，在发送Http/Https协议消息时会携带这些K/V对作为请求header。extension字段允许为空，header字段允许为空。

表 5-168 extension

参数	是否必选	参数类型	描述
header	否	Map<String,String>	header应满足如下要求： 1. key值限定为：包含英文字母([A-Za-z])、数字([0-9])、中划线(-)hyphens，中划线不得作为结尾，不得连续出现。 2. K/V不得超过10个 3. key需要以"x-"开头，不能以"x-smn"开头，正确示例：x-abc-cba, x-abc 4. 所有K/V长度总和不得超过1024个字符 5. key不区分大小写 6. key值不可重复 7. value值限定为ASCII码，不支持中文或其他Unicode，支持空格

响应参数

状态码：200

表 5-169 响应 Body 参数

参数	参数类型	描述
request_id	String	请求ID
task_id	String	探测任务ID

状态码：400

表 5-170 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-171 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-172 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：429

表 5-173 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-174 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

正确调用接口示例

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_create_topic_v2/detect  
  
{  
  "protocol": "http",  
  "endpoint": "http://172.20.0.51:32000/http_notify",  
  "extension": {  
    "header": {  
      "x-abc": "1213",  
      "x-abc-cba": "test2"  
    }  
  }  
}
```

响应示例

状态码：200

发送http探测成功

```
{  
  "request_id": "adc82efa7e0042899722531681b1959a",  
  "task_id": "adc82efa7e0042899722531681b1959a"  
}
```

状态码：400

发送http探测失败

```
{  
  "request_id": "373cdb9696e44454832ea02e61a2c689",  
  "error_code": "SMN.00011004",  
  "error_msg": "Parameter: Extension is invalid. Http header key is invalid."  
}
```

SDK 代码示例

SDK代码示例如下。

Java

正确调用接口示例

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
import java.util.Map;  
import java.util.HashMap;  
  
public class PublishHttpDetectSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

SmnClient client = SmnClient.newBuilder()
    .withCredential(auth)
    .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
    .build();
PublishHttpDetectRequest request = new PublishHttpDetectRequest();
request.withTopicUrn("{topic_urn}");
HttpDetectRequestBody body = new HttpDetectRequestBody();
Map<String, String> listExtensionHeader = new HashMap<>();
listExtensionHeader.put("x-abc", "1213");
listExtensionHeader.put("x-abc-cba", "test2");
HttpDetectRequestBodyExtension extensionbody = new HttpDetectRequestBodyExtension();
extensionbody.withHeader(listExtensionHeader);
body.withExtension(extensionbody);
body.withEndpoint("http://172.20.0.51:32000/http_notify");
body.withProtocol(HttpDetectRequestBody.ProtocolEnum.fromValue("http"));
request.withBody(body);
try {
    PublishHttpDetectResponse response = client.publishHttpDetect(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

正确调用接口示例

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()
```



```
try:
    request = PublishHttpDetectRequest()
    request.topic_urn = "{topic_urn}"
    listHeaderExtension = {
        "x-abc": "1213",
        "x-abc-cba": "test2"
    }
    extensionbody = HttpDetectRequestBodyExtension(
        header=listHeaderExtension
    )
    request.body = HttpDetectRequestBody(
        extension=extensionbody,
        endpoint="http://172.20.0.51:32000/http_notify",
        protocol="http"
    )
    response = client.publish_http_detect(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

正确调用接口示例

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.PublishHttpDetectRequest{
        request.TopicUrn = "{topic_urn}"
    }
    var listHeaderExtension = map[string]string{
        "x-abc": "1213",
        "x-abc-cba": "test2",
    }
    extensionbody := &model.HttpDetectRequestBodyExtension{
        Header: listHeaderExtension,
    }
}
```

```
request.Body = &model.HttpDetectRequestBody{
    Extension: extensionbody,
    Endpoint: "http://172.20.0.51:32000/http_notify",
    Protocol: model.GetHttpDetectRequestBodyProtocolEnum().HTTP,
}
response, err := client.PublishHttpDetect(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	发送http探测成功
400	发送http探测失败
403	Unauthorized
404	Not Found
429	Too Many Requests
500	Internal Server Error

错误码

请参见[错误码](#)。

5.4.3 获取 http 探测结果

功能介绍

根据http探测发送返回的task_id查询探测结果。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-175 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
task_id	是	String	http探测任务id

请求参数

表 5-176 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-177 响应 Body 参数

参数	参数类型	描述
detail	detail object	终端探测详情
status	Integer	http探测任务状态，0代表执行成功，终端可用，1代表未执行，2代表执行失败，终端不可用
request_id	String	请求ID

表 5-178 detail

参数	参数类型	描述
httpCode	Integer	探测终端返回的http返回码，0代表用户在黑名单无法发送，-1代表用户终端超过5秒未响应，-2代表队列已满，Http探测任务未执行。其他httpcode为终端实际返回值。

参数	参数类型	描述
httpResponse	String	终端探测响应体，如果httpCode为0, -1, -2, 2xx时响应体内容固定，由SMN定义。其余httpCode的响应体内容为终端返回内容。

状态码：400

表 5-179 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-180 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-181 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-182 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

正确查询的示例

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_create_topic_v2/detect/  
c414cfd4326a488e9c6a315ce8dc5fe3
```

响应示例

状态码：200

获取http探测正确返回体

```
{  
  "detail" : {  
    "httpCode" : 200,  
    "httpResponse" : "Http endpoint is available."  
  },  
  "status" : 0,  
  "request_id" : "e4bde8f67bd64402804011160c9f8772"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
public class ShowHttpDetectResultSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

SmnClient client = SmnClient.newBuilder()
    .withCredential(auth)
    .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
    .build();
ShowHttpDetectResultRequest request = new ShowHttpDetectResultRequest();
request.withTopicUrn("{topic_urn}");
request.withTaskId("{task_id}");
try {
    ShowHttpDetectResultResponse response = client.showHttpDetectResult(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowHttpDetectResultRequest()
        request.topic_urn = "{topic_urn}"
        request.task_id = "{task_id}"
        response = client.show_http_detect_result(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowHttpDetectResultRequest{}
    request.TopicUrn = "{topic_urn}"
    request.TaskId = "{task_id}"
    response, err := client.ShowHttpDetectResult(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	获取http探测正确返回体
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5 使用标签管理服务

5.5.1 查询资源实例

功能介绍

使用标签过滤实例。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/{resource_type}/resource_instances/action

表 5-183 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
resource_type	是	String	资源类型，目前有： smn_topic，主题 smn_sms，短信 smn_application，移动推送

请求参数

表 5-184 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-185 请求 Body 参数

参数	是否必选	参数类型	描述
tags	否	Array of ResourceTags objects	最多包含10个key，每个key最多包含10个value，结构体不能缺失。key不能为空或者空字符串。key不能重复，同一个key中value不能重复，不同key对应的资源之间为与的关系。
tags_any	否	Array of ResourceTags objects	最多包含10个key，每个key最多包含10个value，结构体不能缺失。key不能为空或者空字符串。key不能重复，同一个key中value不能重复，不同key对应的资源之间为或的关系。
not_tags	否	Array of ResourceTags objects	最多包含10个key，每个key最多包含10个value，结构体不能缺失。key不能为空或者空字符串。key不能重复，同一个key中value不能重复，不同key对应的资源之间为与非的关系。
not_tags_any	否	Array of ResourceTags objects	最多包含10个key，每个key最多包含10个value，结构体不能缺失。key不能为空或者空字符串。key不能重复，同一个key中value不能重复，不同key对应的资源之间为或非的关系。
offset	否	String	索引位置，从offset指定的下一条数据开始查询。查询第一页数据时，不需要传入此参数，查询后续页码数据时，将查询前一页数据时响应体中的值带入此参数。action为count时无此参数。action为filter时，默认为0，必须为数字，且不能为负数。
limit	否	String	查询记录数。 action为count时无此参数。 action为filter时，默认为1000。limit最多为1000，不能为负数，最小值为1。
action	是	String	操作标识（仅限于filter，count）：filter（过滤），count（查询总条数）。 为filter时表示分页查询，为count只需按照条件将总条数返回即可。

参数	是否必选	参数类型	描述
matches	否	Array of TagMatch objects	搜索字段。 key为要匹配的字段，当前只支持resource_name。 value为匹配的值，当前为精确匹配。
without_any_tag	否	Boolean	不包含任意一个标签。该字段为true时查询所有不带标签的资源，此时忽略“tags”、“tags_any”、“not_tags”、“not_tags_any”字段。

表 5-186 ResourceTags

参数	是否必选	参数类型	描述
key	是	String	键。 <ul style="list-style-type: none"> 最大长度127个unicode字符。 key不能为空。
values	是	Array of strings	值列表。 <ul style="list-style-type: none"> 最多10个value。 value不允许重复。 每个值最大长度255个unicode字符。 如果values为空则表示any_value。 value之间为或的关系。

表 5-187 TagMatch

参数	是否必选	参数类型	描述
key	是	String	键，表示要匹配的字段。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> key不能重复，value为匹配的值。 此字段为固定字典值。 不允许为空字符串。

参数	是否必选	参数类型	描述
value	是	String	值。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none">• 每个值最大长度255个unicode字符。• 不可以为空。

响应参数

状态码：200

表 5-188 响应 Body 参数

参数	参数类型	描述
resources	Array of TagResource objects	返回的资源列表。
total_count	Integer	总记录数。

表 5-189 TagResource

参数	参数类型	描述
resource_id	String	资源ID。
resource_detail	ResourceDetail object	资源详情。
tags	Array of ResourceTag objects	标签列表。
resource_name	String	资源名称。

表 5-190 ResourceDetail

参数	参数类型	描述
enterprise_project_id	String	企业项目ID
detailId	String	详情ID

参数	参数类型	描述
topic_urn	String	topic唯一标识
display_name	String	显示名

表 5-191 ResourceTag

参数	参数类型	描述
key	String	键，表示要匹配的字段。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> key不能重复，value为匹配的值。 此字段为固定字典值。 不允许为空字符串。
value	String	值。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> 每个值最大长度255个unicode字符。 不可以为空。

状态码：400

表 5-192 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-193 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。

参数	参数类型	描述
message	String	服务异常错误信息描述。

状态码：404

表 5-194 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-195 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

- 使用标签过滤资源实例

POST https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/resource_instances/action

```
{
  "offset": "100",
  "limit": "100",
  "action": "filter",
  "matches": [ {
    "key": "resource_name",
    "value": "resource1"
  } ],
  "not_tags": [ {
    "key": "key1",
    "values": [ "*"value1", "value2" ]
  }, {
    "key": "key2",
    "values": [ "*"value21", "value22" ]
  } ],
  "tags": [ {
    "key": "key1",
    "values": [ "*"value1", "value2" ]
  } ],
  "tags_any": [ {
    "key": "key1",
```

```
"values" : [ "value1", "value2" ]
}],
"not_tags_any" : [ {
  "key" : "key1",
  "values" : [ "value1", "value2" ]
}]
}
```

- 使用标签查询资源实例总数

POST https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/resource_instances/action

```
{
  "action" : "count",
  "not_tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "*value2" ]
  } ],
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "not_tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource"
  } ]
}
```

响应示例

状态码: 200

OK

```
{
  "resources" : {
    "resource_detail" : {
      "topic_urn" : "urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:resoucece1",
      "display_name" : "testtest",
      "enterprise_project_id" : "0"
    },
    "resource_id" : "cffe4fc4c9a54219b60dbaf7b586e132",
    "resource_name" : "resoucece1",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  } ],
  "total_count" : 1000
}
```

SDK 代码示例

SDK代码示例如下。

Java

- 使用标签过滤资源实例

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListResourceInstancesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        ListResourceInstancesRequest request = new ListResourceInstancesRequest();
        request.withResourceType("{resource_type}");
        ListInstanceRequestBody body = new ListInstanceRequestBody();
        List<TagMatch> listbodyMatches = new ArrayList<>();
        listbodyMatches.add(
            new TagMatch()
                .withKey("resource_name")
                .withValue("resource1")
        );
        List<String> listNotTagsAnyValues = new ArrayList<>();
        listNotTagsAnyValues.add("value1");
        listNotTagsAnyValues.add("value2");
        List<ResourceTags> listbodyNotTagsAny = new ArrayList<>();
        listbodyNotTagsAny.add(
            new ResourceTags()
                .withKey("key1")
                .withValues(listNotTagsAnyValues)
        );
        List<String> listNotTagsValues = new ArrayList<>();
        listNotTagsValues.add("**value21");
        listNotTagsValues.add("value22");
        List<String> listNotTagsValues1 = new ArrayList<>();
        listNotTagsValues1.add("**value1");
        listNotTagsValues1.add("value2");
        List<ResourceTags> listbodyNotTags = new ArrayList<>();
        listbodyNotTags.add(
            new ResourceTags()
                .withKey("key1")
                .withValues(listNotTagsValues1)
        );
        listbodyNotTags.add(
            new ResourceTags()
                .withKey("key2")
                .withValues(listNotTagsValues)
        );
    }
}
```

```
List<String> listTagsAnyValues = new ArrayList<>();
listTagsAnyValues.add("value1");
listTagsAnyValues.add("value2");
List<ResourceTags> listbodyTagsAny = new ArrayList<>();
listbodyTagsAny.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listTagsAnyValues)
);
List<String> listTagsValues = new ArrayList<>();
listTagsValues.add("value1");
listTagsValues.add("value2");
List<ResourceTags> listbodyTags = new ArrayList<>();
listbodyTags.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listTagsValues)
);
body.withMatches(listbodyMatches);
body.withAction("filter");
body.withLimit("100");
body.withOffset("100");
body.withNotTagsAny(listbodyNotTagsAny);
body.withNotTags(listbodyNotTags);
body.withTagsAny(listbodyTagsAny);
body.withTags(listbodyTags);
request.withBody(body);
try {
    ListResourceInstancesResponse response = client.listResourceInstances(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- 使用标签查询资源实例总数

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListResourceInstancesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```



```
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

SmnClient client = SmnClient.newBuilder()
    .withCredential(auth)
    .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
    .build();

ListResourceInstancesRequest request = new ListResourceInstancesRequest();
request.withResourceType("{resource_type}");
ListInstanceRequestBody body = new ListInstanceRequestBody();
List<TagMatch> listbodyMatches = new ArrayList<>();
listbodyMatches.add(
    new TagMatch()
        .withKey("resource_name")
        .withValue("resource")
);
List<String> listNotTagsAnyValues = new ArrayList<>();
listNotTagsAnyValues.add("value1");
listNotTagsAnyValues.add("value2");
List<ResourceTags> listbodyNotTagsAny = new ArrayList<>();
listbodyNotTagsAny.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listNotTagsAnyValues)
);
List<String> listNotTagsValues = new ArrayList<>();
listNotTagsValues.add("value1");
listNotTagsValues.add("value2");
List<ResourceTags> listbodyNotTags = new ArrayList<>();
listbodyNotTags.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listNotTagsValues)
);
List<String> listTagsAnyValues = new ArrayList<>();
listTagsAnyValues.add("value1");
listTagsAnyValues.add("value2");
List<ResourceTags> listbodyTagsAny = new ArrayList<>();
listbodyTagsAny.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listTagsAnyValues)
);
List<String> listTagsValues = new ArrayList<>();
listTagsValues.add("value1");
listTagsValues.add("value2");
List<ResourceTags> listbodyTags = new ArrayList<>();
listbodyTags.add(
    new ResourceTags()
        .withKey("key1")
        .withValues(listTagsValues)
);
body.withMatches(listbodyMatches);
body.withAction("count");
body.withNotTagsAny(listbodyNotTagsAny);
body.withNotTags(listbodyNotTags);
body.withTagsAny(listbodyTagsAny);
body.withTags(listbodyTags);
request.withBody(body);
try {
    ListResourceInstancesResponse response = client.listResourceInstances(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
```

```
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

- 使用标签过滤资源实例

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListResourceInstancesRequest()
        request.resource_type = "{resource_type}"
        listMatchesbody = [
            TagMatch(
                key="resource_name",
                value="resource1"
            )
        ]
        listValuesNotTagsAny = [
            "value1",
            "value2"
        ]
        listNotTagsAnybody = [
            ResourceTags(
                key="key1",
                values=listValuesNotTagsAny
            )
        ]
        listValuesNotTags = [
            "*value21",
            "value22"
        ]
        listValuesNotTags1 = [
            "*value1",
            "value2"
        ]
        listNotTagsbody = [
            ResourceTags(
```

```
        key="key1",
        values=listValuesNotTags1
    ),
    ResourceTags(
        key="key2",
        values=listValuesNotTags
    )
]
listValuesTagsAny = [
    "value1",
    "value2"
]
listTagsAnybody = [
    ResourceTags(
        key="key1",
        values=listValuesTagsAny
    )
]
listValuesTags = [
    "*value1",
    "value2"
]
listTagsbody = [
    ResourceTags(
        key="key1",
        values=listValuesTags
    )
]
request.body = ListInstanceRequestBody(
    matches=listMatchesbody,
    action="filter",
    limit="100",
    offset="100",
    not_tags_any=listNotTagsAnybody,
    not_tags=listNotTagsbody,
    tags_any=listTagsAnybody,
    tags=listTagsbody
)
response = client.list_resource_instances(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- 使用标签查询资源实例总数

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(SmnRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = ListResourceInstancesRequest()
    request.resource_type = "{resource_type}"
    listMatchesbody = [
        TagMatch(
            key="resource_name",
            value="resource"
        )
    ]
    listValuesNotTagsAny = [
        "value1",
        "value2"
    ]
    listNotTagsAnybody = [
        ResourceTags(
            key="key1",
            values=listValuesNotTagsAny
        )
    ]
    listValuesNotTags = [
        "value1",
        "*value2"
    ]
    listNotTagsbody = [
        ResourceTags(
            key="key1",
            values=listValuesNotTags
        )
    ]
    listValuesTagsAny = [
        "value1",
        "value2"
    ]
    listTagsAnybody = [
        ResourceTags(
            key="key1",
            values=listValuesTagsAny
        )
    ]
    listValuesTags = [
        "value1",
        "value2"
    ]
    listTagsbody = [
        ResourceTags(
            key="key1",
            values=listValuesTags
        )
    ]
    request.body = ListInstanceRequestBody(
        matches=listMatchesbody,
        action="count",
        not_tags_any=listNotTagsAnybody,
        not_tags=listNotTagsbody,
        tags_any=listTagsAnybody,
        tags=listTagsbody
    )
    response = client.list_resource_instances(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- 使用标签过滤资源实例

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListResourceInstancesRequest{}
    request.ResourceType = "{resource_type}"
    var listMatchesbody = []model.TagMatch{
        {
            Key: "resource_name",
            Value: "resource1",
        },
    }
    var listValuesNotTagsAny = []string{
        "value1",
        "value2",
    }
    var listNotTagsAnybody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesNotTagsAny,
        },
    }
    var listValuesNotTags = []string{
        "*value21",
        "value22",
    }
    var listValuesNotTags1 = []string{
        "*value1",
        "value2",
    }
    var listNotTagsbody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesNotTags1,
        },
        {
            Key: "key2",
```

```
        Values: listValuesNotTags,
    },
}
var listValuesTagsAny = []string{
    "value1",
    "value2",
}
var listTagsAnybody = []model.ResourceTags{
    {
        Key: "key1",
        Values: listValuesTagsAny,
    },
}
var listValuesTags = []string{
    "value1",
    "value2",
}
var listTagsbody = []model.ResourceTags{
    {
        Key: "key1",
        Values: listValuesTags,
    },
}
limitListInstanceRequestBody:= "100"
offsetListInstanceRequestBody:= "100"
request.Body = &model.ListInstanceRequestBody{
    Matches: &listMatchesbody,
    Action: "filter",
    Limit: &limitListInstanceRequestBody,
    Offset: &offsetListInstanceRequestBody,
    NotTagsAny: &listNotTagsAnybody,
    NotTags: &listNotTagsbody,
    TagsAny: &listTagsAnybody,
    Tags: &listTagsbody,
}
response, err := client.ListResourceInstances(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- 使用标签查询资源实例总数

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
}
```

```
client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListResourceInstancesRequest{
    request.ResourceType = "{resource_type}"
    var listMatchesbody = []model.TagMatch{
        {
            Key: "resource_name",
            Value: "resource",
        },
    }
    var listValuesNotTagsAny = []string{
        "value1",
        "value2",
    }
    var listNotTagsAnybody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesNotTagsAny,
        },
    }
    var listValuesNotTags = []string{
        "value1",
        "*value2",
    }
    var listNotTagsbody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesNotTags,
        },
    }
    var listValuesTagsAny = []string{
        "value1",
        "value2",
    }
    var listTagsAnybody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesTagsAny,
        },
    }
    var listValuesTags = []string{
        "value1",
        "value2",
    }
    var listTagsbody = []model.ResourceTags{
        {
            Key: "key1",
            Values: listValuesTags,
        },
    }
    request.Body = &model.ListInstanceRequestBody{
        Matches: &listMatchesbody,
        Action: "count",
        NotTagsAny: &listNotTagsAnybody,
        NotTags: &listNotTagsbody,
        TagsAny: &listTagsAnybody,
        Tags: &listTagsbody,
    }
    response, err := client.ListResourceInstances(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5.2 批量添加删除资源标签

功能介绍

为指定实例批量添加或删除标签。一个资源上最多有10个标签。

此接口为幂等接口：创建时如果请求体中存在重复key则报错。

创建时，不允许重复key，如果数据库存在就覆盖。

删除时，如果删除的标签不存在，默认处理成功，删除时不对标签字符集范围做校验。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/tags/action

表 5-196 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

参数	是否必选	参数类型	描述
resource_type	是	String	资源类型，目前有： smn_topic，主题 smn_sms，短信 smn_application，移动推送
resource_id	是	String	资源ID。 获取resource_id的方法： 当resource_type为 “smn_topic”时， 手动添加请求消息头“X-SMN- RESOURCEID-TYPE=name”， 资源ID即为topic名称。 不添加请求消息头，通过“查询 资源实例”，获取资源ID。 当resource_type为 “smn_sms”时，resource_id 为签名ID。您可在控制台获取。

请求参数

表 5-197 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户 Token接口获取（响应消息头中 X-Subject-Token的值）。

表 5-198 请求 Body 参数

参数	是否必选	参数类型	描述
tags	是	Array of ResourceTag objects	标签列表，结构体说明请参见表 1。删除时tags结构体不能缺失，key不能为空或空字符串，且不针对字符集范围进行校验。
action	是	String	操作标识：仅限于create（创建）、delete（删除）。

表 5-199 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	键，表示要匹配的字段。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> key不能重复，value为匹配的值。 此字段为固定字典值。 不允许为空字符串。
value	是	String	值。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> 每个值最大长度255个unicode字符。 不可以为空。

响应参数

状态码：400

表 5-200 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-201 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-202 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-203 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

- 批量添加资源标签

POST https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags/action

```
{
  "action": "create",
  "tags": [ {
    "key": "key1",
    "value": "value1"
  }, {
    "key": "key",
    "value": "value3"
  } ]
}
```

- 批量删除资源标签

POST https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags/action

```
{
  "action": "delete",
  "tags": [ {
    "key": "key1"
  }, {
    "key": "key2",
    "value": "value3"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

- 批量添加资源标签

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchCreateOrDeleteResourceTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchCreateOrDeleteResourceTagsRequest request = new
        BatchCreateOrDeleteResourceTagsRequest();
        request.withResourceType("{resource_type}");
        request.withResourceId("{resource_id}");
        BatchCreateOrDeleteResourceTagsRequestBody body = new
        BatchCreateOrDeleteResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key")
                .withValue("value3")
        );
        body.withAction("create");
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            BatchCreateOrDeleteResourceTagsResponse response =
            client.batchCreateOrDeleteResourceTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- 批量删除资源标签

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchCreateOrDeleteResourceTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchCreateOrDeleteResourceTagsRequest request = new
BatchCreateOrDeleteResourceTagsRequest();
        request.withResourceType("{resource_type}");
        request.withResourceId("{resource_id}");
        BatchCreateOrDeleteResourceTagsRequestBody body = new
BatchCreateOrDeleteResourceTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key2")
                .withValue("value3")
        );
        body.withAction("delete");
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
```

```
        BatchCreateOrDeleteResourceTagsResponse response =
client.batchCreateOrDeleteResourceTags(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

- 批量添加资源标签

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchCreateOrDeleteResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1",
                value="value1"
            ),
            ResourceTag(
                key="key",
                value="value3"
            )
        ]
        request.body = BatchCreateOrDeleteResourceTagsRequestBody(
            action="create",
            tags=listTagsbody
        )
        response = client.batch_create_or_delete_resource_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

- 批量删除资源标签

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchCreateOrDeleteResourceTagsRequest()
        request.resource_type = "{resource_type}"
        request.resource_id = "{resource_id}"
        listTagsbody = [
            ResourceTag(
                key="key1"
            ),
            ResourceTag(
                key="key2",
                value="value3"
            )
        ]
        request.body = BatchCreateOrDeleteResourceTagsRequestBody(
            action="delete",
            tags=listTagsbody
        )
        response = client.batch_create_or_delete_resource_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- 批量添加资源标签

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchCreateOrDeleteResourceTagsRequest{}
    request.ResourceType = "{resource_type}"
    request.ResourceId = "{resource_id}"
    var listTagsbody = []model.ResourceTag{
        {
            Key: "key1",
            Value: "value1",
        },
        {
            Key: "key",
            Value: "value3",
        },
    }
    request.Body = &model.BatchCreateOrDeleteResourceTagsRequestBody{
        Action: "create",
        Tags: listTagsbody,
    }
    response, err := client.BatchCreateOrDeleteResourceTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- 批量删除资源标签

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```



```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.BatchCreateOrDeleteResourceTagsRequest{}
request.ResourceType = "{resource_type}"
request.ResourceId = "{resource_id}"
var listTagsbody = []model.ResourceTag{
    {
        Key: "key1",
    },
    {
        Key: "key2",
        Value: "value3",
    },
}
request.Body = &model.BatchCreateOrDeleteResourceTagsRequestBody{
    Action: "delete",
    Tags: listTagsbody,
}
response, err := client.BatchCreateOrDeleteResourceTags(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5.3 查询资源标签

功能介绍

查询指定实例的标签信息。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/{resource_type}/{resource_id}/tags

表 5-204 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
resource_type	是	String	资源类型 目前有： smn_topic，主题 smn_sms，短信 smn_application，移动推送
resource_id	是	String	资源ID。 获取resource_id的方法： 当resource_type为 “smn_topic”时， 手动添加请求消息头“X-SMN- RESOURCEID-TYPE=name”， 资源ID即为topic名称。 不添加请求消息头，通过“ 查询资源实例 ”，获取资源ID。 当resource_type为 “smn_sms”时，resource_id 为签名ID。您可在控制台获取。

请求参数

表 5-205 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-206 响应 Body 参数

参数	参数类型	描述
tags	Array of ResourceTag objects	资源标签列表。

表 5-207 ResourceTag

参数	参数类型	描述
key	String	键，表示要匹配的字段。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none">key不能重复，value为匹配的值。此字段为固定字典值。不允许为空字符串。
value	String	值。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none">每个值最大长度255个unicode字符。不可以为空。

状态码：400

表 5-208 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-209 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-210 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-211 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询资源标签

```
GET https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags  
null
```

响应示例

状态码：200

OK

```
{  
  "tags": [ {  
    "key": "key1",  
    "value": "value1"  
  }, {  
    "key": "key2",  
    "value": "value3"  
  } ]  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5.4 添加资源标签

功能介绍

一个资源上最多有10个标签。此接口为幂等接口：创建时，如果创建的标签已经存在（key相同），则覆盖。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/{resource_type}/{resource_id}/tags

表 5-212 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
resource_type	是	String	资源类型，目前有： smn_topic，主题 smn_sms，短信 smn_application，移动推送
resource_id	是	String	资源ID。 获取resource_id的方法： 当resource_type为 “smn_topic”时， 手动添加请求消息头“X-SMN- RESOURCEID-TYPE=name”， 资源ID即为topic名称。 不添加请求消息头，通过“ 查询资源实例 ”，获取资源ID。 当resource_type为 “smn_sms”时，resource_id 为签名ID。您可在控制台获取。

请求参数

表 5-213 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-214 请求 Body 参数

参数	是否必选	参数类型	描述
tag	是	ResourceTag object	资源标签结构体。

表 5-215 ResourceTag

参数	是否必选	参数类型	描述
key	是	String	键，表示要匹配的字段。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> key不能重复，value为匹配的值。 此字段为固定字典值。 不允许为空字符串。
value	是	String	值。 当前key的参数值只能取“resource_name”，此时value的参数值为云服务器名称。 <ul style="list-style-type: none"> 每个值最大长度255个unicode字符。 不可以为空。

响应参数

状态码：400

表 5-216 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-217 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-218 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-219 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

添加资源标签

```
POST https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags
{
  "tag": {
    "key": "DEV",
    "value": "DEV1"
  }
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

添加资源标签

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```



```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class CreateResourceTagSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateResourceTagRequest request = new CreateResourceTagRequest();
        request.withResourceType("{resource_type}");
        request.withResourceId("{resource_id}");
        CreateResourceTagRequestBody body = new CreateResourceTagRequestBody();
        ResourceTag tagbody = new ResourceTag();
        tagbody.withKey("DEV")
            .withValue("DEV1");
        body.withTag(tagbody);
        request.withBody(body);
        try {
            CreateResourceTagResponse response = client.createResourceTag(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

添加资源标签

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = SmnClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(SmnRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateResourceTagRequest()
    request.resource_type = "{resource_type}"
    request.resource_id = "{resource_id}"
    tagbody = ResourceTag(
        key="DEV",
        value="DEV1"
    )
    request.body = CreateResourceTagRequestBody(
        tag=tagbody
    )
    response = client.create_resource_tag(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

添加资源标签

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateResourceTagRequest{}
```

```
request.ResourceType = "{resource_type}"
request.ResourceId = "{resource_id}"
tagbody := &model.ResourceTag{
    Key: "DEV",
    Value: "DEV1",
}
request.Body = &model.CreateResourceTagRequestBody{
    Tag: tagbody,
}
response, err := client.CreateResourceTag(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5.5 查询项目标签

功能介绍

查询租户在指定Region和实例类型的所有标签集合。

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/{resource_type}/tags

表 5-220 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
resource_type	是	String	资源类型 目前有： smn_topic, 主题 smn_sms, 短信 smn_application, 移动推送

请求参数

表 5-221 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-222 响应 Body 参数

参数	参数类型	描述
tags	Array of ResourceTags objects	资源标签列表。

表 5-223 ResourceTags

参数	参数类型	描述
key	String	键。 <ul style="list-style-type: none">最大长度127个unicode字符。key不能为空。

参数	参数类型	描述
values	Array of strings	值列表。 <ul style="list-style-type: none"> • 最多10个value。 • value不允许重复。 • 每个值最大长度255个unicode字符。 • 如果values为空则表示any_value。 • value之间为或的关系。

状态码：400

表 5-224 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-225 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-226 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-227 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

查询项目标签

```
GET https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/tags
```

```
null
```

响应示例

状态码：200

OK

```
{  
  "tags": [ {  
    "key": "key1",  
    "values": [ "value1", "value2" ]  
  }, {  
    "key": "key2",  
    "values": [ "value1", "value2" ]  
  } ]  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.5.6 删除资源标签

功能介绍

幂等接口：删除时，不对标签做校验。删除的key不存在报404，key不能为空或者空字符串。

调用方法

请参见[如何调用API](#)。

URI

DELETE /v2/{project_id}/{resource_type}/{resource_id}/tags/{key}

表 5-228 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
resource_type	是	String	资源类型，目前有： smn_topic，主题。 smn_sms，短信。 smn_application，移动推送。
resource_id	是	String	资源ID。 获取resource_id的方法： 当resource_type为 “smn_topic”时， 手动添加请求消息头“X-SMN-RESOURCEID-TYPE=name”， 资源ID即为topic名称。 不添加请求消息头，通过“查询资源实例”，获取资源ID。 当resource_type为 “smn_sms”时，resource_id 为签名ID。您可在控制台获取。
key	是	String	待删除标签的key值。

请求参数

表 5-229 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：400

表 5-230 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-231 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-232 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-233 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除资源标签

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/{resource_type}/{resource_id}/tags/{key}
null
```

响应示例

无

状态码

状态码	描述
204	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.6 查询版本操作

5.6.1 查询 SMN 支持的 API 版本号信息

功能介绍

查询SMN开放API支持的版本号。

调用方法

请参见[如何调用API](#)。

URI

GET /

请求参数

表 5-234 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-235 响应 Body 参数

参数	参数类型	描述
versions	Array of VersionItem objects	描述version相关对象的列表。

表 5-236 VersionItem

参数	参数类型	描述
id	String	版本ID（版本号），如v2。
min_version	String	若该版本API支持微版本，则返回支持的最小微版本号；若不支持微版本，则返回空。
status	String	版本状态，为如下3种： CURRENT：表示该版本为主推版本。 SUPPORTED：表示为老版本，但是现在还继续支持。 DEPRECATED：表示为废弃版本，存在后续删除的可能。

参数	参数类型	描述
updated	String	版本发布时间，要求用UTC时间格式表示。如v2发布的时间2014-06-28T12:20:21Z。
version	String	若该版本API支持微版本，则返回支持的最大微版本号；若不支持微版本，则返回空。
links	Array of LinkItem objects	API的URL地址。

表 5-237 LinkItem

参数	参数类型	描述
href	String	对应快捷链接。
rel	String	快捷链接标记名称。

状态码：400

表 5-238 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-239 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-240 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-241 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/
null
```

响应示例

状态码：200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request

状态码	描述
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.6.2 查询 SMN API V2 版本信息

功能介绍

查询SMN API V2版本信息。

调用方法

请参见[如何调用API](#)。

URI

GET /v2

请求参数

表 5-242 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-243 响应 Body 参数

参数	参数类型	描述
version	VersionItem object	描述V2相关对象的列表。

表 5-244 VersionItem

参数	参数类型	描述
id	String	版本ID（版本号），如v2。
min_version	String	若该版本API支持微版本，则返回支持的最小微版本号；若不支持微版本，则返回空。
status	String	版本状态，为如下3种： CURRENT：表示该版本为主推版本。 SUPPORTED：表示为老版本，但是现在还继续支持。 DEPRECATED：表示为废弃版本，存在后续删除的可能。
updated	String	版本发布时间，要求用UTC时间格式表示。如v2发布的时间 2014-06-28T12:20:21Z。
version	String	若该版本API支持微版本，则返回支持的最大微版本号；若不支持微版本，则返回空。
links	Array of LinkItem objects	API的URL地址。

表 5-245 LinkItem

参数	参数类型	描述
href	String	对应快捷链接。
rel	String	快捷链接标记名称。

状态码：400

表 5-246 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-247 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-248 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-249 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/v2
null
```

响应示例

状态码：200

OK

```
{
  "version": {
    "id": "v2",
    "links": [ {
      "href": "https://127.0.0.1/v2",
      "rel": "self"
    } ],
    "min_version": ""
  }
}
```

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.7 云日志操作

5.7.1 绑定云日志

功能介绍

为指定Topic绑定一个云日志，用于记录主题消息发送状态等信息。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-250 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-251 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-252 请求 Body 参数

参数	是否必选	参数类型	描述
log_group_id	是	String	云日志服务日志组ID。
log_stream_id	是	String	云日志服务日志流ID。

响应参数

状态码：201

表 5-253 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识。
id	String	云日志信息唯一的资源标识。

状态码：400

表 5-254 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-255 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-256 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-257 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

绑定云日志

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/logtanks
{
  "log_group_id" : "d1b6c352-ee45-4196-9230-c74b7894e356",
  "log_stream_id" : "114960b4-1c26-48dd-93c2-4caae24aec49"
}
```

响应示例

状态码：201

OK

```
{
  "request_id" : "42f3b0cd1c894e87aa0275ce33305b21",
```

```
"id": "4ea80c8a68b840e19b4efd3a8f0d62ed"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

绑定云日志

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
public class CreateLogtankSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        SmnClient client = SmnClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateLogtankRequest request = new CreateLogtankRequest();  
        request.withTopicUrn("{topic_urn}");  
        CreateLogtankRequestBody body = new CreateLogtankRequestBody();  
        body.withLogStreamId("114960b4-1c26-48dd-93c2-4caae24aec49");  
        body.withLogGroupId("d1b6c352-ee45-4196-9230-c74b7894e356");  
        request.withBody(body);  
        try {  
            CreateLogtankResponse response = client.createLogtank(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

绑定云日志

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLogtankRequest()
        request.topic_urn = "{topic_urn}"
        request.body = CreateLogtankRequestBody(
            log_stream_id="114960b4-1c26-48dd-93c2-4caae24aec49",
            log_group_id="d1b6c352-ee45-4196-9230-c74b7894e356"
        )
        response = client.create_logtank(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

绑定云日志

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateLogtankRequest{}
request.TopicUrn = "{topic_urn}"
request.Body = &model.CreateLogtankRequestBody{
    LogStreamId: "114960b4-1c26-48dd-93c2-4caae24aec49",
    LogGroupId: "d1b6c352-ee45-4196-9230-c74b7894e356",
}
response, err := client.CreateLogtank(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.7.2 查询云日志

功能介绍

查询指定Topic绑定的云日志。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-258 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-259 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-260 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识
count	Integer	云日志信息数量
logtanks	Array of LogtankItem objects	云日志信息列表

表 5-261 LogtankItem

参数	参数类型	描述
id	String	云日志信息唯一的资源标识。
log_group_id	String	云日志服务日志组ID。

参数	参数类型	描述
log_stream_id	String	云日志服务日志流ID。
create_time	String	创建时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
update_time	String	更新时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。

状态码：400

表 5-262 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-263 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-264 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-265 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

查询云日志

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/logtanks
null
```

响应示例

状态码：200

OK

```
{
  "request_id": "f5c05cb70d264ca58d1080202ceba284",
  "count": 1,
  "logtanks": [ {
    "id": "4ea80c8a68b840e19b4efd3a8f0d62ed",
    "log_group_id": "d1b6c352-ee45-4196-9230-c74b7894e356",
    "log_stream_id": "114960b4-1c26-48dd-93c2-4caae24aec49",
    "create_time": "2022-10-10T09:00:00Z",
    "update_time": "2022-10-10T09:00:00Z"
  } ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.7.3 更新云日志

功能介绍

更新指定Topic绑定的云日志。

调用方法

请参见[如何调用API](#)。

URI

PUT /v2/{project_id}/notifications/topics/{topic_urn}/logtanks/{logtank_id}

表 5-266 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
logtank_id	是	String	云日志信息唯一的资源标识。可通过 查询云日志 获取该标识。

请求参数

表 5-267 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-268 请求 Body 参数

参数	是否必选	参数类型	描述
log_group_id	是	String	云日志服务日志组ID。
log_stream_id	是	String	云日志服务日志流ID。

响应参数

状态码：200

表 5-269 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识。
logtank	LogtankItem object	云日志信息

表 5-270 LogtankItem

参数	参数类型	描述
id	String	云日志信息唯一的资源标识。
log_group_id	String	云日志服务日志组ID。
log_stream_id	String	云日志服务日志流ID。
create_time	String	创建时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
update_time	String	更新时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。

状态码：400

表 5-271 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-272 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。

参数	参数类型	描述
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-273 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-274 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

更新云日志

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/logtanks/
4ea80c8a68b840e19b4efd3a8f0d62ed
```

```
{
  "log_group_id" : "d1b6c352-ee45-4196-9230-c74b7894e356",
  "log_stream_id" : "114960b4-1c26-48dd-93c2-4caae24aec49"
}
```

响应示例

状态码：200

OK

```
{
  "request_id" : "761bd2658f92411b9b01ae13e7961b11",
  "logtank" : {
    "id" : "4ea80c8a68b840e19b4efd3a8f0d62ed",
    "log_group_id" : "d1b6c352-ee45-4196-9230-c74b7894e356",
    "log_stream_id" : "114960b4-1c26-48dd-93c2-4caae24aec49",
  }
}
```

```
"create_time" : "2022-10-10T09:00:00Z",  
"update_time" : "2022-10-10T09:00:00Z"  
}  
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新云日志

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
public class UpdateLogtankSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        SmnClient client = SmnClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateLogtankRequest request = new UpdateLogtankRequest();  
        request.withTopicUrn("{topic_urn}");  
        request.withLogtankId("{logtank_id}");  
        UpdateLogtankRequestBody body = new UpdateLogtankRequestBody();  
        body.withLogStreamId("114960b4-1c26-48dd-93c2-4caae24aec49");  
        body.withLogGroupId("d1b6c352-ee45-4196-9230-c74b7894e356");  
        request.withBody(body);  
        try {  
            UpdateLogtankResponse response = client.updateLogtank(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

```
}  
}
```

Python

更新云日志

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdksmn.v2 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = SmnClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = UpdateLogtankRequest()  
        request.topic_urn = "{topic_urn}"  
        request.logtank_id = "{logtank_id}"  
        request.body = UpdateLogtankRequestBody(  
            log_stream_id="114960b4-1c26-48dd-93c2-4caae24aec49",  
            log_group_id="d1b6c352-ee45-4196-9230-c74b7894e356"  
        )  
        response = client.update_logtank(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

更新云日志

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateLogtankRequest{}
request.TopicUrn = "{topic_urn}"
request.LogtankId = "{logtank_id}"
request.Body = &model.UpdateLogtankRequestBody{
    LogStreamId: "114960b4-1c26-48dd-93c2-4caae24aec49",
    LogGroupId: "d1b6c352-ee45-4196-9230-c74b7894e356",
}
response, err := client.UpdateLogtank(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.7.4 解绑云日志

功能介绍

解绑指定Topic绑定的云日志。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-275 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
logtank_id	是	String	云日志信息唯一的资源标识。可通过 查询云日志 获取该标识。

请求参数

表 5-276 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-277 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识。

状态码：400

表 5-278 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-279 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-280 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-281 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

解绑云日志


```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/logtanks/  
4ea80c8a68b840e19b4efd3a8f0d62ed  
null
```

响应示例

状态码：200

OK

```
{  
  "request_id" : "fe5cec832a5249169d433b6b0b5902a3"  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.8 订阅过滤策略操作

5.8.1 批量创建订阅过滤策略

功能介绍

创建订阅者的消息过滤策略。

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/notifications/subscriptions/filter_policies

表 5-282 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

请求参数

表 5-283 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-284 请求 Body 参数

参数	是否必选	参数类型	描述
policies	是	Array of policies objects	批量处理订阅者策略列表。

表 5-285 policies

参数	是否必选	参数类型	描述
subscription_u rn	是	String	订阅者的唯一的资源标识，可通过 查询订阅者列表 获取该标识。
filter_policies	是	Array of Subscriptions FilterPolicy objects	订阅者的过滤策略。策略name不能重复

表 5-286 SubscriptionsFilterPolicy

参数	是否必选	参数类型	描述
name	是	String	过滤策略名称。 包含小写英文字母([a-z])、数字([0-9])、下划线(_), 下划线不得开始、结尾或连续出现), 长度限制{1,32}, 不能是smn_开头。
string_equals	是	Array of strings	字符串精确匹配数组。数组长度[1, 10], 数组内容不能重复, 值不能为null或者空字符串“”, 长度限制[1,32], 中英文、数字、下划线

响应参数

状态码：200

表 5-287 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
batch_result	Array of BatchResult objects	批量结果

表 5-288 BatchResult

参数	参数类型	描述
code	String	执行结果code
message	String	执行结果message
subscription_urn	String	订阅者urn

状态码：400

表 5-289 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-290 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-291 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-292 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

创建订阅消息过滤策略

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/subscriptions/filter_policies
{
```

```
"policies" : [ {
  "subscription_urn" :
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837
",
  "filter_policies" : [ {
    "name" : "alarm",
    "string_equals" : [ "os", "process" ]
  }, {
    "name" : "service",
    "string_equals" : [ "api", "db" ]
  } ]
} ]
}
```

响应示例

状态码：200

OK

```
{
  "request_id" : "be368401641b406d8c28a79915ba3589",
  "batch_result" : [ {
    "code" : "SMN.00011027",
    "message" : "Parameter: subscription_urn is invalid.",
    "subscription_urn" :
"urn:smn:regionId:98386b0630aa41d990d4729497fcd7ba:test:90b22be1efab4cd6924703c5b228e59f"
  }, {
    "code" : "SMN.00011027",
    "message" : "Parameter: subscription_urn is invalid.",
    "subscription_urn" :
"urn:smn:regionId:98386b0630aa41d990d4729497fcd7ba:test:c872b769e60d45f682f1da44eb4dbec3"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建订阅消息过滤策略

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchCreateSubscriptionsFilterPoliciesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

SmnClient client = SmnClient.newBuilder()
    .withCredential(auth)
    .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
    .build();

BatchCreateSubscriptionsFilterPoliciesRequest request = new
BatchCreateSubscriptionsFilterPoliciesRequest();
BatchSubscriptionsFilterPoliciesRequestBody body = new BatchSubscriptionsFilterPoliciesRequestBody();
List<String> listFilterPoliciesStringEquals = new ArrayList<>();
listFilterPoliciesStringEquals.add("api");
listFilterPoliciesStringEquals.add("db");
List<String> listFilterPoliciesStringEquals1 = new ArrayList<>();
listFilterPoliciesStringEquals1.add("os");
listFilterPoliciesStringEquals1.add("process");
List<SubscriptionsFilterPolicy> listPoliciesFilterPolicies = new ArrayList<>();
listPoliciesFilterPolicies.add(
    new SubscriptionsFilterPolicy()
        .withName("alarm")
        .withStringEquals(listFilterPoliciesStringEquals1)
);
listPoliciesFilterPolicies.add(
    new SubscriptionsFilterPolicy()
        .withName("service")
        .withStringEquals(listFilterPoliciesStringEquals)
);
List<BatchSubscriptionsFilterPoliciesRequestBodyPolicies> listbodyPolicies = new ArrayList<>();
listbodyPolicies.add(
    new BatchSubscriptionsFilterPoliciesRequestBodyPolicies()
        .withSubscriptionUrn("urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e77
8e84408e44058e6cbc6d3c377837")
        .withFilterPolicies(listPoliciesFilterPolicies)
);
body.withPolicies(listbodyPolicies);
request.withBody(body);
try {
    BatchCreateSubscriptionsFilterPoliciesResponse response =
client.batchCreateSubscriptionsFilterPolicies(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

创建订阅消息过滤策略

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
```

```
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchCreateSubscriptionsFilterPoliciesRequest()
        listStringEqualsFilterPolicies = [
            "api",
            "db"
        ]
        listStringEqualsFilterPolicies1 = [
            "os",
            "process"
        ]
        listFilterPoliciesPolicies = [
            SubscriptionsFilterPolicy(
                name="alarm",
                string_equals=listStringEqualsFilterPolicies1
            ),
            SubscriptionsFilterPolicy(
                name="service",
                string_equals=listStringEqualsFilterPolicies
            )
        ]
        listPoliciesbody = [
            BatchSubscriptionsFilterPoliciesRequestBodyPolicies(
                subscription_urn="urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058
                e6cbc6d3c377837",
                filter_policies=listFilterPoliciesPolicies
            )
        ]
        request.body = BatchSubscriptionsFilterPoliciesRequestBody(
            policies=listPoliciesbody
        )
        response = client.batch_create_subscriptions_filter_policies(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建订阅消息过滤策略

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchCreateSubscriptionsFilterPoliciesRequest{}
    var listStringEqualsFilterPolicies = []string{
        "api",
        "db",
    }
    var listStringEqualsFilterPolicies1 = []string{
        "os",
        "process",
    }
    var listFilterPoliciesPolicies = []model.SubscriptionsFilterPolicy{
        {
            Name: "alarm",
            StringEquals: listStringEqualsFilterPolicies1,
        },
        {
            Name: "service",
            StringEquals: listStringEqualsFilterPolicies,
        },
    }
    var listPoliciesbody = []model.BatchSubscriptionsFilterPoliciesRequestBodyPolicies{
        {
            SubscriptionUrn:
                "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837",
            FilterPolicies: listFilterPoliciesPolicies,
        },
    }
    request.Body = &model.BatchSubscriptionsFilterPoliciesRequestBody{
        Policies: listPoliciesbody,
    }
    response, err := client.BatchCreateSubscriptionsFilterPolicies(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.8.2 批量更新订阅过滤策略

功能介绍

更新订阅者的消息过滤策略。

调用方法

请参见[如何调用API](#)。

URI

PUT /v2/{project_id}/notifications/subscriptions/filter_policies

表 5-293 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

请求参数

表 5-294 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-295 请求 Body 参数

参数	是否必选	参数类型	描述
polices	是	Array of polices objects	批量处理订阅者策略列表。

表 5-296 polices

参数	是否必选	参数类型	描述
subscription_u rn	是	String	订阅者的唯一的资源标识, 可通过 查询订阅者列表 获取该标识。
filter_polices	是	Array of Subscriptions FilterPolicy objects	订阅者的过滤策略。策略name不能重复

表 5-297 SubscriptionsFilterPolicy

参数	是否必选	参数类型	描述
name	是	String	过滤策略名称。 包含小写英文字母([a-z])、数字([0-9])、下划线(_), 下划线不得开始、结尾或连续出现), 长度限制{1,32}, 不能是smn_开头。
string_equals	是	Array of strings	字符串精确匹配数组。数组长度[1, 10], 数组内容不能重复, 值不能为null或者空字符串“”, 长度限制[1,32], 中英文、数字、下划线

响应参数

状态码：200

表 5-298 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
batch_result	Array of BatchResult objects	批量结果

表 5-299 BatchResult

参数	参数类型	描述
code	String	执行结果code
message	String	执行结果message
subscription_urn	String	订阅者urn

状态码：400

表 5-300 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-301 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-302 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-303 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新订阅消息过滤策略

PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/subscriptions/filter_policies

```
{
  "policies": [ {
    "subscription_urn":
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837",
    "filter_policies": [ {
      "name": "alarm",
      "string_equals": [ "os", "process" ]
    }, {
      "name": "service",
      "string_equals": [ "api", "db" ]
    } ]
  } ]
}
```

响应示例

状态码：200

OK

```
{
  "request_id": "be368401641b406d8c28a79915ba3589",
  "batch_result": [ {
    "code": "SMN.00011027",
    "message": "Parameter: subscription_urn is invalid.",
    "subscription_urn":
"urn:smn:regionId:98386b0630aa41d990d4729497fcd7ba:test:90b22be1efab4cd6924703c5b228e59f"
  }, {
    "code": "SMN.00011027",
    "message": "Parameter: subscription_urn is invalid.",
    "subscription_urn":
"urn:smn:regionId:98386b0630aa41d990d4729497fcd7ba:test:c872b769e60d45f682f1da44eb4dbee3"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

更新订阅消息过滤策略

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchUpdateSubscriptionsFilterPoliciesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();

        BatchUpdateSubscriptionsFilterPoliciesRequest request = new
        BatchUpdateSubscriptionsFilterPoliciesRequest();
        BatchSubscriptionsFilterPoliciesRequestBody body = new BatchSubscriptionsFilterPoliciesRequestBody();
        List<String> listFilterPoliciesStringEquals = new ArrayList<>();
        listFilterPoliciesStringEquals.add("api");
        listFilterPoliciesStringEquals.add("db");
        List<String> listFilterPoliciesStringEquals1 = new ArrayList<>();
        listFilterPoliciesStringEquals1.add("os");
        listFilterPoliciesStringEquals1.add("process");
        List<SubscriptionsFilterPolicy> listPoliciesFilterPolicies = new ArrayList<>();
        listPoliciesFilterPolicies.add(
            new SubscriptionsFilterPolicy()
                .withName("alarm")
                .withStringEquals(listFilterPoliciesStringEquals1)
        );
        listPoliciesFilterPolicies.add(
            new SubscriptionsFilterPolicy()
                .withName("service")
                .withStringEquals(listFilterPoliciesStringEquals)
        );
        List<BatchSubscriptionsFilterPoliciesRequestBodyPolicies> listbodyPolicies = new ArrayList<>();
        listbodyPolicies.add(
            new BatchSubscriptionsFilterPoliciesRequestBodyPolicies()
                .withSubscriptionUrn("urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e77
                8e84408e44058e6cbc6d3c377837")
        );
    }
}
```

```
        .withFilterPolicies(listPoliciesFilterPolicies)
    );
    body.withPolicies(listbodyPolicies);
    request.withBody(body);
    try {
        BatchUpdateSubscriptionsFilterPoliciesResponse response =
client.batchUpdateSubscriptionsFilterPolicies(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

更新订阅消息过滤策略

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchUpdateSubscriptionsFilterPoliciesRequest()
        listStringEqualsFilterPolicies = [
            "api",
            "db"
        ]
        listStringEqualsFilterPolicies1 = [
            "os",
            "process"
        ]
        listFilterPoliciesPolicies = [
            SubscriptionsFilterPolicy(
                name="alarm",
                string_equals=listStringEqualsFilterPolicies1
            ),
            SubscriptionsFilterPolicy(
                name="service",
                string_equals=listStringEqualsFilterPolicies
            )
        ]
    except exceptions.ApiError as e:
        print(e)
```

```
)
]
listPoliciesbody = [
    BatchSubscriptionsFilterPoliciesRequestBodyPolicies(
subscription_urn="urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058
e6cbc6d3c377837",
        filter_policies=listFilterPoliciesPolicies
    )
]
request.body = BatchSubscriptionsFilterPoliciesRequestBody(
    policies=listPoliciesbody
)
response = client.batch_update_subscriptions_filter_policies(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

更新订阅消息过滤策略

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchUpdateSubscriptionsFilterPoliciesRequest{}
    var listStringEqualsFilterPolicies = []string{
        "api",
        "db",
    }
    var listStringEqualsFilterPolicies1 = []string{
        "os",
        "process",
    }
    var listFilterPoliciesPolicies = []model.SubscriptionsFilterPolicy{
        {
            Name: "alarm",
```

```
StringEquals: listStringEqualsFilterPolicies1,
},
{
  Name: "service",
  StringEquals: listStringEqualsFilterPolicies,
},
}
var listPoliciesbody = []model.BatchSubscriptionsFilterPoliciesRequestBodyPolicies{
  {
    SubscriptionUrn:
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837",
    FilterPolicies: listFilterPoliciesPolicies,
  },
}
request.Body = &model.BatchSubscriptionsFilterPoliciesRequestBody{
  Policies: listPoliciesbody,
}
response, err := client.BatchUpdateSubscriptionsFilterPolicies(request)
if err == nil {
  fmt.Printf("%+v\n", response)
} else {
  fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.8.3 批量删除订阅过滤策略

功能介绍

删除订阅者的消息过滤策略。

调用方法

请参见[如何调用API](#)。

URI

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

表 5-304 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

请求参数

表 5-305 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-306 请求 Body 参数

参数	是否必选	参数类型	描述
subscription_urls	是	Array of strings	订阅者的唯一的资源标识，可通过 查询订阅者列表 获取该标识。

响应参数

状态码：200

表 5-307 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
batch_result	Array of BatchResult objects	批量结果

表 5-308 BatchResult

参数	参数类型	描述
code	String	执行结果code
message	String	执行结果message
subscription_urn	String	订阅者urn

状态码：400

表 5-309 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：403

表 5-310 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：404

表 5-311 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码：500

表 5-312 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除订阅消息过滤策略

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/subscriptions/filter_policies
{
  "subscription_urns" :
  [ "urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837" ]
}
```

响应示例

状态码：200

OK

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

SDK 代码示例

SDK代码示例如下。

Java

删除订阅消息过滤策略

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchDeleteSubscriptionsFilterPoliciesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
```

```
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

SmnClient client = SmnClient.newBuilder()
    .withCredential(auth)
    .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
    .build();
BatchDeleteSubscriptionsFilterPoliciesRequest request = new
BatchDeleteSubscriptionsFilterPoliciesRequest();
BatchDeleteSubscriptionsFilterPoliciesRequestBody body = new
BatchDeleteSubscriptionsFilterPoliciesRequestBody();
List listbodySubscriptionUrns = new ArrayList<>();

listbodySubscriptionUrns.add("urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e8
4408e44058e6cbc6d3c377837");
body.withSubscriptionUrns(listbodySubscriptionUrns);
request.withBody(body);
try {
    BatchDeleteSubscriptionsFilterPoliciesResponse response =
client.batchDeleteSubscriptionsFilterPolicies(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

删除订阅消息过滤策略

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
```

```
.build()

try:
    request = BatchDeleteSubscriptionsFilterPoliciesRequest()
    listSubscriptionUrnsbody = [
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837
"
    ]
    request.body = BatchDeleteSubscriptionsFilterPoliciesRequestBody(
        subscription_urns=listSubscriptionUrnsbody
    )
    response = client.batch_delete_subscriptions_filter_policies(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

删除订阅消息过滤策略

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchDeleteSubscriptionsFilterPoliciesRequest{}
    var listSubscriptionUrnsbody = []string{
"urn:smn:regionId:762bdb3251034f268af0e395c53ea09b:test_topic_v1:2e778e84408e44058e6cbc6d3c377837
",
    }
    request.Body = &model.BatchDeleteSubscriptionsFilterPoliciesRequestBody{
        SubscriptionUrns: listSubscriptionUrnsbody,
    }
    response, err := client.BatchDeleteSubscriptionsFilterPolicies(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
```

```
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.9 通知策略

5.9.1 创建通知策略

功能介绍

创建通知策略

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/notifications/topics/{topic_urn}/notify-policy

表 5-313 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。

参数	是否必选	参数类型	描述
topic_urn	是	String	Topic的唯一的资源标识, 可通过 查询主题列表 获取该标识。

请求参数

表 5-314 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-315 请求 Body 参数

参数	是否必选	参数类型	描述
protocol	是	String	通知策略类型, 当前仅支持语音。
polling	是	Array of PollingPolicyRequest objects	轮询策略订阅终端。

表 5-316 PollingPolicyRequest

参数	是否必选	参数类型	描述
order	是	Integer	当前轮询的序号。
subscription_urns	是	Array of strings	订阅终端urn列表。

响应参数

状态码：201

表 5-317 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID

参数	参数类型	描述
id	String	通知策略ID

状态码：400

表 5-318 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-319 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-320 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-321 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

创建轮询的通知策略。

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/notify-policy
```

```
{  
  "protocol": "callnotify",  
  "polling": [ {  
    "order": 1,  
    "subscription_urns": [  
      "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",  
      "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222" ]  
    }, {  
      "order": 2,  
      "subscription_urns": [  
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",  
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222" ]  
      } ]  
}
```

响应示例

状态码：201

创建成功

```
{  
  "request_id": "f6832d5dd17f4090b0e29c6f9a026b17",  
  "id": "7824d9c2f4e94fdabd9fcdd4da0d564"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

创建轮询的通知策略。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;  
import com.huaweicloud.sdk.smn.v2.*;  
import com.huaweicloud.sdk.smn.v2.model.*;  
  
import java.util.List;  
import java.util.ArrayList;
```

```
public class CreateNotifyPolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateNotifyPolicyRequest request = new CreateNotifyPolicyRequest();
        request.withTopicUrn("{topic_urn}");
        NotifyPolicyRequestBody body = new NotifyPolicyRequestBody();
        List<String> listPollingSubscriptionUrns = new ArrayList<>();

        listPollingSubscriptionUrns.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f");

        listPollingSubscriptionUrns.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222");
        List<String> listPollingSubscriptionUrns1 = new ArrayList<>();

        listPollingSubscriptionUrns1.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f");

        listPollingSubscriptionUrns1.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222");
        List<PollingPolicyRequest> listbodyPolling = new ArrayList<>();
        listbodyPolling.add(
            new PollingPolicyRequest()
                .withOrder(1)
                .withSubscriptionUrns(listPollingSubscriptionUrns1)
        );
        listbodyPolling.add(
            new PollingPolicyRequest()
                .withOrder(2)
                .withSubscriptionUrns(listPollingSubscriptionUrns)
        );
        body.withPolling(listbodyPolling);
        body.withProtocol(NotifyPolicyRequestBody.ProtocolEnum.fromValue("callnotify"));
        request.withBody(body);
        try {
            CreateNotifyPolicyResponse response = client.createNotifyPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

创建轮询的通知策略。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateNotifyPolicyRequest()
        request.topic_urn = "{topic_urn}"
        listSubscriptionUrnsPolling = [

"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",

"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222"
        ]
        listSubscriptionUrnsPolling1 = [

"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",

"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222"
        ]
        listPollingbody = [
            PollingPolicyRequest(
                order=1,
                subscription_urns=listSubscriptionUrnsPolling1
            ),
            PollingPolicyRequest(
                order=2,
                subscription_urns=listSubscriptionUrnsPolling
            )
        ]
        request.body = NotifyPolicyRequestBody(
            polling=listPollingbody,
            protocol="callnotify"
        )
        response = client.create_notify_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

创建轮询的通知策略。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateNotifyPolicyRequest{}
    request.TopicUrn = "{topic_urn}"
    var listSubscriptionUrnsPolling = []string{
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    }
    var listSubscriptionUrnsPolling1 = []string{
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    }
    var listPollingbody = []model.PollingPolicyRequest{
        {
            Order: int32(1),
            SubscriptionUrns: listSubscriptionUrnsPolling1,
        },
        {
            Order: int32(2),
            SubscriptionUrns: listSubscriptionUrnsPolling,
        },
    }
    request.Body = &model.NotifyPolicyRequestBody{
        Polling: listPollingbody,
        Protocol: model.GetNotifyPolicyRequestBodyProtocolEnum().CALLNOTIFY,
    }
    response, err := client.CreateNotifyPolicy(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	创建成功
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.9.2 查询通知策略

功能介绍

查询通知策略

调用方法

请参见[如何调用API](#)。

URI

GET /v2/{project_id}/notifications/topics/{topic_urn}/notify-policy

表 5-322 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

请求参数

表 5-323 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-324 响应 Body 参数

参数	参数类型	描述
request_id	String	请求ID。
id	String	通知策略ID。
protocol	String	通知策略类型，当前仅支持语音。
polling	Array of PollingPolicyResponse objects	轮询策略订阅终端。

表 5-325 PollingPolicyResponse

参数	参数类型	描述
order	Integer	当前轮询的序号。
subscriptions	Array of PollingPolicySubscriptionDetails objects	订阅终端urn列表。

表 5-326 PollingPolicySubscriptionDetails

参数	参数类型	描述
subscription_urn	String	当前轮询的序号。
endpoint	String	订阅终端urn列表。
remark	String	备注。

参数	参数类型	描述
status	Integer	订阅者状态：0表示订阅还未确认，1表示已经确认，3表示已经取消确认。

状态码：403

表 5-327 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-328 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

查询通知策略。

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/urn:smn:regionId:f96188c7ccaf4ffb0c9aa149ab2bd57:test_topic_v2/notify-policy
```

响应示例

状态码：200

查询成功

```
{
  "request_id": "2f97a871015348529bccc070417ef16f",
  "protocol": "callnotify",
  "id": "9d1ff67d195d44c1aed4edd92cc44cd1",
  "polling": [ {
    "order": 1,
    "subscriptions": [ {
      "subscription_urn": "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
      "endpoint": "13612345611",
      "remark": ""
    }
  ]
}
```

```
    "status" : 1
  }, {
    "subscription_urn" :
"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    "endpoint" : "13612345622",
    "remark" : "",
    "status" : 1
  }
  ]
}, {
  "order" : 2,
  "subscriptions" : [ {
    "subscription_urn" :
"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
    "endpoint" : "13612345633",
    "remark" : "",
    "status" : 1
  }, {
    "subscription_urn" :
"urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    "endpoint" : "13612345644",
    "remark" : "",
    "status" : 1
  }
  ]
}
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class ShowNotifyPolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowNotifyPolicyRequest request = new ShowNotifyPolicyRequest();
        request.withTopicUrn("{topic_urn}");
        try {
```



```
        ShowNotifyPolicyResponse response = client.showNotifyPolicy(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowNotifyPolicyRequest()
        request.topic_urn = "{topic_urn}"
        response = client.show_notify_policy(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := smn.NewSmnClient(
    smn.SmnClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowNotifyPolicyRequest{}
request.TopicUrn = "{topic_urn}"
response, err := client.ShowNotifyPolicy(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	查询成功
403	Unauthorized
500	Internal Server Error

错误码

请参见[错误码](#)。

5.9.3 修改通知策略

功能介绍

修改通知策略，该接口仅支持全量修改，不支持部分修改。

调用方法

请参见[如何调用API](#)。

URI

PUT /v2/{project_id}/notifications/topics/{topic_urn}/notify-policy/
{notify_policy_id}

表 5-329 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。
notify_policy_id	是	String	通知策略ID。

请求参数

表 5-330 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 5-331 请求 Body 参数

参数	是否必选	参数类型	描述
protocol	是	String	通知策略类型，当前仅支持语音。
polling	是	Array of PollingPolicyRequest objects	轮询策略订阅终端。

表 5-332 PollingPolicyRequest

参数	是否必选	参数类型	描述
order	是	Integer	当前轮询的序号。
subscription_urns	是	Array of strings	订阅终端urn列表。

响应参数

状态码：200

表 5-333 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID

状态码：400

表 5-334 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：403

表 5-335 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-336 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-337 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

修改通知策略。

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/  
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/notify-policy/  
7824d9c2f4e94fdfabd9fcdd4da0d564  
  
{  
  "protocol": "callnotify",  
  "polling": [ {  
    "order": "1",  
    "subscription_urns":  
    [ "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",  
      "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222" ]  
    }, {  
    "order": "2",  
    "subscription_urns":  
    [ "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",  
      "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222" ]  
    } ]  
}
```

响应示例

状态码：200

修改成功

```
{  
  "request_id": "f6832d5dd17f4090b0e29c6f9a026b17"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

修改通知策略。

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateNotifyPolicySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        SmnClient client = SmnClient.newBuilder()
            .withCredential(auth)
            .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateNotifyPolicyRequest request = new UpdateNotifyPolicyRequest();
        request.withTopicUrn("{topic_urn}");
        request.withNotifyPolicyId("{notify_policy_id}");
        NotifyPolicyRequestBody body = new NotifyPolicyRequestBody();
        List<String> listPollingSubscriptionUrns = new ArrayList<>();

        listPollingSubscriptionUrns.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f");

        listPollingSubscriptionUrns.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222");
        List<String> listPollingSubscriptionUrns1 = new ArrayList<>();

        listPollingSubscriptionUrns1.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f");

        listPollingSubscriptionUrns1.add("urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222");
        List<PollingPolicyRequest> listbodyPolling = new ArrayList<>();
        listbodyPolling.add(
            new PollingPolicyRequest()
                .withOrder(1)
                .withSubscriptionUrns(listPollingSubscriptionUrns1)
        );
        listbodyPolling.add(
            new PollingPolicyRequest()
                .withOrder(2)
                .withSubscriptionUrns(listPollingSubscriptionUrns)
        );
        body.withPolling(listbodyPolling);
        body.withProtocol(NotifyPolicyRequestBody.ProtocolEnum.fromValue("callnotify"));
        request.withBody(body);
        try {
            UpdateNotifyPolicyResponse response = client.updateNotifyPolicy(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

修改通知策略。

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudskmsmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudskmsmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateNotifyPolicyRequest()
        request.topic_urn = "{topic_urn}"
        request.notify_policy_id = "{notify_policy_id}"
        listSubscriptionUrnsPolling = [
            "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
            "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222"
        ]
        listSubscriptionUrnsPolling1 = [
            "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
            "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222"
        ]
        listPollingbody = [
            PollingPolicyRequest(
                order=1,
                subscription_urns=listSubscriptionUrnsPolling1
            ),
            PollingPolicyRequest(
                order=2,
                subscription_urns=listSubscriptionUrnsPolling
            )
        ]
        request.body = NotifyPolicyRequestBody(
            polling=listPollingbody,
            protocol="callnotify"
        )
        response = client.update_notify_policy(request)
        print(response)
```

```
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

修改通知策略。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateNotifyPolicyRequest{}
    request.TopicUrn = "{topic_urn}"
    request.NotifyPolicyId = "{notify_policy_id}"
    var listSubscriptionUrnsPolling = []string{
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    }
    var listSubscriptionUrnsPolling1 = []string{
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b521f",
        "urn:smn:regionId:240489644ad14ace8e022869d34411a4:eee:45c05e14762c478aa2ef6b16153b5222",
    }
    var listPollingbody = []model.PollingPolicyRequest{
        {
            Order: int32(1),
            SubscriptionUrns: listSubscriptionUrnsPolling1,
        },
        {
            Order: int32(2),
            SubscriptionUrns: listSubscriptionUrnsPolling,
        },
    }
    request.Body = &model.NotifyPolicyRequestBody{
        Polling: listPollingbody,
        Protocol: model.GetNotifyPolicyRequestBodyProtocolEnum().CALLNOTIFY,
    }
    response, err := client.UpdateNotifyPolicy(request)
    if err == nil {
```



```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	修改成功
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

5.9.4 删除通知策略

功能介绍

删除通知策略

调用方法

请参见[如何调用API](#)。

URI

DELETE /v2/{project_id}/notifications/topics/{topic_urn}/notify-policy/{notify_policy_id}

表 5-338 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取项目ID请参考 获取项目ID 。
topic_urn	是	String	Topic的唯一的资源标识，可通过 查询主题列表 获取该标识。

参数	是否必选	参数类型	描述
notify_policy_id	是	String	通知策略ID。

请求参数

表 5-339 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码：200

表 5-340 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID

状态码：403

表 5-341 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：404

表 5-342 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

状态码：500

表 5-343 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
error_code	String	返回信息对应的代码。
error_msg	String	服务异常错误信息描述。

请求示例

删除通知策略。

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/
urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2/notify-policy/
7824d9c2f4e94dfabd9fcdd4da0d564
```

响应示例

状态码：200

删除成功

```
{
  "request_id" : "f6832d5dd17f4090b0e29c6f9a026b17"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.smn.v2.region.SmnRegion;
import com.huaweicloud.sdk.smn.v2.*;
import com.huaweicloud.sdk.smn.v2.model.*;

public class DeleteNotifyPolicySolution {
```

```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");
    String projectId = "{project_id}";

    ICredential auth = new BasicCredentials()
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

    SmnClient client = SmnClient.newBuilder()
        .withCredential(auth)
        .withRegion(SmnRegion.valueOf("<YOUR REGION>"))
        .build();
    DeleteNotifyPolicyRequest request = new DeleteNotifyPolicyRequest();
    request.withTopicUrn("{topic_urn}");
    request.withNotifyPolicyId("{notify_policy_id}");
    try {
        DeleteNotifyPolicyResponse response = client.deleteNotifyPolicy(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdksmn.v2.region.smn_region import SmnRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdksmn.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = SmnClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(SmnRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteNotifyPolicyRequest()
        request.topic_urn = "{topic_urn}"
```

```
request.notify_policy_id = "{notify_policy_id}"
response = client.delete_notify_policy(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    smn "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/smn/v2/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := smn.NewSmnClient(
        smn.SmnClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteNotifyPolicyRequest{}
    request.TopicUrn = "{topic_urn}"
    request.NotifyPolicyId = "{notify_policy_id}"
    response, err := client.DeleteNotifyPolicy(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	删除成功

状态码	描述
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

6 应用示例

- 6.1 示例1: 使用主题发布消息
- 6.2 示例2: 使用消息模板方式发布消息

6.1 示例 1: 使用主题发布消息

操作场景

本章节指导用户通过API发布消息。API的调用方法请参见[如何调用API](#)。

涉及接口

- **获取用户Token**: 通过用户名/密码的方式进行认证来获取IAM用户Token。
- **创建主题**: 创建Topic。
- **订阅**: 为指定Topic添加订阅者。
- **消息发布**: 将消息发送给Topic的所有订阅终端。

操作步骤

1. Token认证，具体操作请参考[认证鉴权](#)。
2. 创建Topic。
 - a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics”。
 - b. 在Request Header中增加“X-Auth-Token”。
 - c. 在Request Body中传入如下参数：

详细的参数说明请参见[创建主题](#)。

```
{
  "name": "test_topic_v2", //Topic名称（必填String）
  "display_name": "test", //显示名（非必填String）
  "enterprise_project_id": "0" //企业项目ID(企业项目场景下为必填String)
}
```

响应示例:

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
```

```
"topic_urn": "urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2"
}
```

3. 添加订阅。

- a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/subscriptions”。

其中，topic_urn可在2.c的返回体中获取。

- b. 在Request Header中增加“X-Auth-Token”。
- c. 在Request Body中传入参数如下。

```
{
  "protocol": "email", //订阅协议（必填String）
  "endpoint": "xxx@xx.com", //邮件地址（必填String）
  "remark": "订阅终端" //备注（非必填String）
}
```

响应示例：

```
{
  "subscriptions_result": [{
    "request_id": "126c17f805934e5eadf647b0860c7a0a",
    "subscription_urn": "urn:smn:regionId:f96188c7ccaf4ffba0c9aa149ab2bd57:test_topic_v2",
    "http_code": 201}]
}
```

说明

可为Topic添加多种以及多个订阅终端，支持的订阅协议请参考订阅。

- d. 订阅终端会收到一条确认订阅的消息，请在邮箱中进行确认订阅的操作。

4. 发送消息。

- a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/publish”。

- b. 在Request Header中增加“X-Auth-Token”。
- c. 在Request Body中传入参数如下：

```
{
  "subject": "test message v2", //标题，邮件订阅终端使用（非必填String）
  "message": "Message test message v2." //需要发送的消息（必填String）
}
```

响应示例：

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "message_id": "992bbd32819448af95f83814cb3c7454"
}
```

说明

已完成3.c的确认订阅操作的订阅终端都能接收到该消息，否则无法接收消息。

6.2 示例 2：使用消息模板方式发布消息

操作场景

本章节指导用户通过API使用消息模板方式发布消息。API的调用方法请参见[如何调用API](#)。

涉及接口

- **获取用户Token**：通过用户名/密码的方式进行认证来获取IAM用户Token。

- **创建主题**：创建Topic。
- **订阅**：为指定Topic添加订阅者。
- **创建消息模板**：创建一个模板，用户可以按照模板去发送消息，这样可以减少请求的数据量。
- **消息发布**：将消息发送给Topic的所有订阅终端。

操作步骤

1. Token认证，具体操作请参考[认证鉴权](#)。
2. 创建Topic。
 - a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics”。
 - b. 在Request Header中增加“X-Auth-Token”。
 - c. 在Request Body中传入如下参数。
详细的参数说明请参见[创建主题](#)。

```
{
  "name": "publish_by_template", //Topic名称（必填String）
  "display_name": "test", //显示名（非必填String）
  "enterprise_project_id": "0" //企业项目ID(企业项目场景下为必填String)
}
```

响应示例：

```
{
  "request_id": "a8f8161449564f569c19e2a9e3be25c9",
  "topic_urn": "urn:smn:br-iaas-odin1:0605767e0a00d5762fe7c00d0e33345c:publish_by_template"
}
```
3. 添加订阅。
 - a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/subscriptions”。其中，topic_urn可在2.c的返回体中获取。
 - b. 在Request Header中增加“X-Auth-Token”。
 - c. 在Request Body中传入参数如下。

```
{
  "protocol": "sms", //订阅协议（必填String）
  "endpoint": "86137****3629", //邮件地址（必填String）
  "remark": "订阅终端" //备注（非必填String）
}
```

响应示例：

```
{
  "subscriptions_result": [{
    "request_id": "126c17f805934e5eadf647b0860c7a0a",
    "subscription_urn": "urn:smn:br-iaas-odin1:0605767e0a00d5762fe7c00d0e33345c:publish_by_template:3e5ac4544a1947b7b0d9f743fe52bdb",
    "http_code": 201
  }]
}
```
4. 创建消息模板。
发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/message_template”。

说明

可为Topic添加多种以及多个订阅终端，支持的订阅协议请参考[订阅](#)。

- d. 订阅终端会收到一条确认订阅的消息，请在邮箱中进行确认订阅的操作。

5. 在Request Header中增加“X-Auth-Token”。

a. 在Request Body中传入参数如下：

```
{
  "message_template_name": "sms_template_test",
  "locale": "zh-cn",
  "content": "{content}",
  "protocol": "sms"
}
```

响应示例：

```
{
  "request_id": "e67228ac92324a58980e922ca600f8ef",
  "message_template_id": "cdb113996c5d49149fb97af36d9801ce"
}
```

b. 订阅终端会收到一条确认订阅的消息，请在手机中进行确认订阅的操作。

6. 使用消息模板方式发布消息。

a. 发送“POST https://{SMN_Endpoint}/v2/{project_id}/notifications/topics/{topic_urn}/publish”。

b. 在Request Header中增加“X-Auth-Token”。

c. 在Request Body中传入参数如下：

```
{
  "message_template_name": "sms_template_test",
  "tags": {
    "content": "hello test"
  },
  "subject": "template message test"
}
```

响应示例：

```
{
  "message_id": "5da965fd9e2c4036a0542317d944af83",
  "request_id": "c6be4517f985470e8448e42f9d2ec5a2"
}
```

 说明

已完成3.c的确认订阅操作的订阅终端都能接收到该消息，否则无法接收消息。

7 权限和授权项

- 7.1 权限及授权项说明
- 7.2 主题操作
- 7.3 模板操作
- 7.4 标签操作
- 7.5 发布消息操作
- 7.6 Application操作（废弃）
- 7.7 Application直发消息操作（废弃）

7.1 权限及授权项说明

如果您需要对您所拥有的消息通知服务（Simple Message Notification）进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称IAM），如果华为云账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用SMN服务的其它功能。

默认情况下，新建的IAM用户没有任何权限，您需要将其加入用户组，并给用户组授予策略或角色，才能使用户组中的用户获得相应的权限，这一过程称为授权。授权后，用户就可以基于已有权限对云服务进行操作。

权限根据授权的精细程度，分为**角色**和**策略**。角色以服务为粒度，是IAM最初提供了一种根据用户的工作职能定义权限的粗粒度授权机制。策略以API接口为粒度进行权限拆分，授权更加精细，可以精确到某个操作、资源和条件，能够满足企业对权限最小化的安全管控要求。

说明

如果您要允许或是禁止某个接口的操作权限，请使用策略。

账号具备所有接口的调用权限，如果使用账号下的IAM用户发起API请求时，该IAM用户必须具备调用该接口所需的权限，否则，API请求将调用失败。每个接口所需要的权限，与各个接口所对应的授权项相对应，只有发起请求的用户被授予授权项所对应的策略，该用户才能成功调用该接口。例如，用户要调用接口来查询云服务器列表，那么这个IAM用户被授予的策略中必须包含允许“ecs:servers:list”的授权项，该接口才能调用成功。

支持的授权项

策略包含系统策略和自定义策略，如果系统策略不满足授权要求，管理员可以创建自定义策略，并通过给用户组授予自定义策略来进行精细的访问控制。策略支持的操作与API相对应，授权项列表说明如下：

- 权限：允许或拒绝某项操作。
- 对应API接口：自定义策略实际调用的API接口。
- 授权项：自定义策略中支持的Action，在自定义策略中的Action中写入授权项，可以实现授权项对应的权限功能。
- 依赖的授权项：部分Action存在对其他Action的依赖，需要将依赖的Action同时写入授权项，才能实现对应的权限功能。
- IAM项目(Project)/企业项目(Enterprise Project)：自定义策略的授权范围，包括IAM项目与企业项目。授权范围如果同时支持IAM项目和企业项目，表示此授权项对应的自定义策略，可以在IAM和企业管理两个服务中给用户组授权并生效。如果仅支持IAM项目，不支持企业项目，表示仅能在IAM中给用户组授权并生效，如果在企业管理中授权，则该自定义策略不生效。管理员可以在授权项列表中查看授权项是否支持IAM项目或企业项目，“√”表示支持，“×”表示暂不支持。关于IAM项目与企业项目的区别，详情请参见：[IAM与企业管理的区别](#)。

消息通知服务（SMN）的支持自定义策略授权项如下所示：

- [7.2 主题操作](#)，包含Topic管理接口对应的授权项，如创建主题、查询主题、更新主题、删除主题。
- [消息模板操作](#)，包含消息模板管理接口对应的授权项，如创建消息模板、查询消息模板、更新消息模板、删除消息模板等接口。
- [Application操作](#)，包含Application操作管理接口对应的授权项，如创建应用、查询应用、更新应用、删除应用等接口。
- [使用标签管理服务](#)，包含使用标签管理服务管理接口对应的授权项，如查询资源实例、添加资源标签、批量添加删除资源标签、删除资源标签等接口。
- [发布消息操作](#)，包含发布消息操作管理接口对应的授权项，如消息发布接口。
- [7.7 Application直发消息操作（废弃）](#)，包含Application直发消息操作管理接口对应的授权项，如发布应用消息接口。

7.2 主题操作

权限	对应API接口	授权项	IAM项目	企业项目
创建主题	POST /v2/{project_id}/notifications/topics	smn:topic:create	√	√
查询主题	GET /v2/{project_id}/notifications/topics	smn:topic:list	√	√

权限	对应API接口	授权项	IAM项目	企业项目
更新主题	PUT /v2/ {project_id}/ notifications/ topics/ {topic_urn}	smn:topic:upd ate	√	√
删除主题	DELETE /v2/ {project_id}/ notifications/ topics/ {topic_urn}	smn:topic:dele te	√	√

7.3 模板操作

权限	对应API接口	授权项	IAM项目	企业项目
创建消息模板	POST /v2/ {project_id}/ notifications/ message_tem plate	smn:template: create	√	×
查询消息模板	GET /v2/ {project_id}/ notifications/ message_tem plate	smn:template: list	√	×
更新消息模板	PUT /v2/ {project_id}/ notifications/ message_tem plate/ {message_te mplate_id}	smn:template: update	√	×
删除消息模板	DELETE /v2/ {project_id}/ notifications/ message_tem plate/ {message_te mplate_id}	smn:template: delete	√	×

7.4 标签操作

权限	对应API接口	授权项	IAM项目	企业项目
添加资源标签	POST /v2/{project_id}/{resource_type}/{resource_id}/tags	smn:tag:create	√	√
更新标签	-	smn:tag:update	√	√
删除资源标签	DELETE /v2/{project_id}/{resource_type}/{resource_id}/tags/{key}	smn:tag:delete	√	√
查询资源标签	GET /v2/{project_id}/{resource_type}/{resource_id}/tags	smn:tag:list	√	√

7.5 发布消息操作

权限	对应API接口	授权项	IAM项目	企业项目
发布消息	POST /v2/{project_id}/notifications/topics/{topic_urn}/publish	smn:topic:publish	√	×

7.6 Application 操作（废弃）

权限	对应API接口	授权项	IAM项目	企业项目
创建应用	POST /v2/{project_id}/notifications/applications	smn:application:create	√	√
查询应用	GET /v2/{project_id}/notifications/applications?offset={offset}&limit={limit}&name={name}&platform={platform}	smn:application:list	√	√
更新应用	PUT /v2/{project_id}/notifications/applications/{application_urn}	smn:application:update	√	√
删除应用	DELETE /v2/{project_id}/notifications/applications/{application_urn}	smn:application:delete	√	√

7.7 Application 直发消息操作（废弃）

权限	对应API接口	授权项	IAM项目	企业项目
发布应用消息	POST /v2/{project_id}/notifications/endpoints/{endpoint_urn}/publish	smn:application:publish	√	√

8 公共参数

8.1 获取项目ID

8.2 application消息体格式（废弃）

8.1 获取项目 ID

操作场景

在调用接口的时候，部分URL中需要填入项目ID，所以需要获取到项目ID。有如下两种获取方式：

- [调用API获取项目ID](#)
- [从控制台获取项目ID](#)

调用 API 获取项目 ID

项目ID可以通过调用[查询指定条件下的项目列表](#)API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects”，其中{Endpoint}为IAM的终端节点，可以从[地区和终端节点](#)获取。接口的认证鉴权请参见[3.2 认证鉴权](#)。

响应示例如下，其中projects下的“id”即为项目ID。

```
{
  "projects": [
    {
      "domain_id": "65ewtrgaggshhk1223245sghjlse684b",
      "is_domain": false,
      "parent_id": "65ewtrgaggshhk1223245sghjlse684b",
      "name": "project_name",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4adasfjljaaakla12334jklga9sasfg"
      },
      "id": "a4adasfjljaaakla12334jklga9sasfg",
      "enabled": true
    }
  ],
  "links": {
```



```
"next": null,  
"previous": null,  
"self": "https://www.example.com/v3/projects"  
}  
}
```

从控制台获取项目 ID

从控制台获取项目ID的步骤如下：

1. 登录管理控制台。
2. 鼠标悬停在右上角的用户名，选择下拉列表中的“我的凭证”。
在“API凭证”页面的项目列表中查看项目ID。

图 8-1 查看项目 ID



8.2 application 消息体格式（废弃）

表 8-1 application 消息体格式

类型	样例	备注
华为透传消息	<pre>"hps": { "msg" : { "type" : 1, "body" : {"key": "value"} } }</pre>	<p>body为用户自定义格式。</p> <p>说明</p> <ul style="list-style-type: none">• 推荐使用JSON格式，如样例所示• 对于纯文本可以将body设置为字符串(采用UTF-8编码)• 消息体最大2K 详细的字段介绍及其使用请参见华为推送平台的接口定义。

类型	样例	备注
华为系统通知栏消息	<pre>"hps": { "msg": { "type": 3, "body": { "content": "Push message content", "title": "Push message content" }, "action": { "type": 1, "param": { "intent": "#Intent;compo=com.rvr/.Activity;S.W=U;end" } } }, "ext": { "biTag": "Trump", "icon": "http://upload.w.org/00/150pxsvg.png" } }</pre>	
苹果平台消息格式	<pre>{ "aps": { "alert": "hello world" } }</pre>	<p>消息体最大4K。 详细的字段介绍及其使用参见苹果推送平台的接口定义。</p>

9 历史 API

- [9.1 Application操作](#)
- [9.2 Application endpoint操作](#)
- [9.3 Application直发消息操作](#)
- [9.4 订阅确认V2接口（废弃）](#)

9.1 Application 操作

9.1.1 查询 Application（废弃）

功能介绍

查询应用平台列表。

URI

GET /v2/{project_id}/notifications/applications

表 9-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。

表 9-2 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量。 偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。 缺省值： 0
limit	否	Integer	查询的数量限制。 取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。 缺省值： 100
name	否	String	检索应用名称。 支持后向模糊匹配，长度不得超过64个字符，只能包含英文字母、下划线和数字。
platform	否	String	应用平台。 目前仅支持HMS、APNS、APNS_SANDBOX。 HMS是为开发者提供的消息推送平台。 APNS和APNS_SANDBOX是用于推送iOS消息的服务平台。

请求参数

表 9-3 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-4 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
application_count	Integer	返回的Application个数。该参数不受offset和limit影响，即返回的是您账户下所有的Application个数。
applications	Array of ApplicationItem objects	Application列表。

表 9-5 ApplicationItem

参数	参数类型	描述
name	String	创建application的名字。
platform	String	应用平台。
create_time	String	创建application的时间。时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。
application_url	String	Application的唯一资源标识。
application_id	String	Application的唯一标识ID。
enabled	String	应用平台是否启用。
apple_certificate_expiration_date	String	苹果证书过期时间APNS、APNS_SANDBOX平台特有属性时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。

状态码： 400

表 9-6 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-7 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-8 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-9 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/applications?
offset=0&limit=100&name=application_name&platform=APNS
null
```

响应示例

状态码： 200

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "application_count": 2,
  "applications": [ {
    "create_time": "2018-02-08T08:30:16Z",
    "apple_certificate_expiration_date": "2018-03-09T12:21:40Z",
    "name": "application_name001",
```

```

    "platform": "APNS",
    "enabled": "true",
    "application_urn": "urn:smn:regionId:429ffced18074da0938112f2c362b935:app-APNS-
application_name001",
    "application_id": "b1b8643dc12b4g77ad6e35a16003119b"
  }, {
    "create_time": "2018-01-11T12:58:58Z",
    "apple_certificate_expiration_date": "2018-03-09T12:21:40Z",
    "name": "application_name002",
    "platform": "APNS",
    "enabled": "true",
    "application_urn": "urn:smn:regionId:429ffced18074da0938112f2c362b935:app-APNS-
application_name002",
    "application_id": "a3a4643dc12b4g77ad6e35a16002558c"
  }
]
}

```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.1.2 创建 Application (废弃)

功能介绍

创建平台应用。

URI

POST /v2/{project_id}/notifications/applications

表 9-10 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。

请求参数

表 9-11 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 9-12 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用名。 最大支持64个字符，只能包含英文字母、下划线和数字。
platform	是	String	应用平台。 目前仅支持HMS、APNS、APNS_SANDBOX。 HMS是为开发者提供的消息推送平台。 APNS和APNS_SANDBOX是用于推送iOS消息的服务平台。
platform_principal	是	String	对于HMS平台是APP ID，只能包含英文字母和数字，最大20个字符。对于苹果APNS、APNS_SandBox平台是推送证书，大小不超过8K，且是Base64编码。
platform_credential	是	String	对于HMS平台是APP SECRET，只能包含英文字母和数字，32到64个字符。 对于苹果APNS、APNS_SandBox平台是推送证书的私钥（private key），大小不超过8K，且是Base64编码。

响应参数

状态码： 201

表 9-13 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
application_url	String	Application的唯一资源标识。
application_id	String	Application资源的ID。

状态码： 400

表 9-14 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-15 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-16 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-17 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

创建Application

POST https://{SMN_Endpoint}/v2/{project_id}/notifications/applications

```
{
  "name" : "application_name",
  "platform" : "HMS",
  "platform_principal" : "appId",
  "platform_credential" : "appSecret"
}
```

响应示例

状态码： 201

OK

```
{
  "request_id" : "6a63a18b8bab40ffb71ebd9cb80d0085",
  "application_urn" : "urn:smn:regionId:429ffced18074da0938112f2c362b935:app-APNS-
application_name002",
  "application_id" : "a3a4643dc12b4g77ad6e35a16002558c"
}
```

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.1.3 查询 Application 属性（废弃）

功能介绍

获取应用平台属性。

URI

GET /v2/{project_id}/notifications/applications/{application_urn}

表 9-18 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
application_urn	是	String	Application的唯一资源标识，可通过查询Application获取该标识。

请求参数

表 9-19 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-20 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
application_id	String	Application的唯一标识ID。
attributes	attributes object	Application的属性列表。

表 9-21 attributes

参数	参数类型	描述
enabled	String	应用平台是否启用。
apple_certificate_expiration_date	String	苹果证书过期时间，APNS、APNS_SANDBOX平台特有属性 时间格式为UTC时间，YYYY-MM-DDTHH:MM:SSZ。

状态码： 400

表 9-22 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-23 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-24 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-25 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}
null
```

响应示例

状态码： 200

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "application_id": "b1b8643dc12b4g77ad6e35a16003119b",
  "attributes": {
    "enabled": "true",
    "apple_certificate_expiration_date": "2018-03-09T12:21:40Z"
  }
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.1.4 更新 Application（废弃）

功能介绍

更新应用平台。

URI

PUT /v2/{project_id}/notifications/applications/{application_urn}

表 9-26 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
application_urn	是	String	Application的唯一资源标识， 可通过查询Application获取该标识。

请求参数

表 9-27 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 9-28 请求 Body 参数

参数	是否必选	参数类型	描述
platform_principal	是	String	对于HMS平台是APP ID，只能包含英文字母和数字，最大20个字符。 对于苹果APNS、APNS_SandBox平台是推送证书，大小不超过8K，且是Base64编码。
platform_credential	是	String	对于HMS平台是APP SECRET，只能包含英文字母和数字，32到64个字符。 对于苹果APNS、APNS_SandBox平台是推送证书的私钥（private key），大小不超过8K，且是Base64编码。

响应参数

状态码： 200

表 9-29 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码： 400

表 9-30 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-31 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-32 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-33 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新应用平台

```
PUT https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}
{
  "platform_principal": "appld",
  "platform_credential": "appSecret"
}
```

响应示例

状态码： 200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.1.5 删除 Application（废弃）

功能介绍

删除平台应用。

URI

DELETE /v2/{project_id}/notifications/applications/{application_urn}

表 9-34 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
application_urn	是	String	Application的唯一资源标识，可通过查询Application获取该标识。

请求参数

表 9-35 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-36 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码： 400

表 9-37 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。

参数	参数类型	描述
message	String	服务异常错误信息描述。

状态码： 403

表 9-38 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-39 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-40 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除平台应用

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}
{
  "platform_principal": "appld",
```

```
"platform_credential" : "appSecret"  
}
```

响应示例

无

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.2 Application endpoint 操作

9.2.1 查询 Application 的 Endpoint 列表（废弃）

功能介绍

查询平台的endpoint列表。

URI

GET /v2/{project_id}/notifications/applications/{application_urn}/endpoints

表 9-41 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
application_urn	是	String	Application的唯一资源标识，可通过查询Application获取该标识。

表 9-42 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量。 偏移量为一个大于0小于资源总个数的整数，表示查询该偏移量后面的所有的资源，默认值为0。 缺省值： 0
limit	否	Integer	查询的数量限制。 取值范围：1~100，取值一般为10，20，50。功能说明：每页返回的资源个数。默认值为100。 缺省值： 100
enabled	否	String	设备是否可用，值为true或false字符串。
token	否	String	设备token，最大长度512个字节。
user_data	否	String	用户数据，最大长度2048个字节。

请求参数

表 9-43 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-44 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
next_page_flag	Boolean	是否有下一页标识。
endpoints	Array of ApplicationEndpoint objects	Application_endpoint结构体数。

表 9-45 ApplicationEndpoint

参数	参数类型	描述
create_time	String	创建application的时间 时间格式为UTC时间, YYYY-MM-DDTHH:MM:SSZ。
endpoint_urn	String	Application endpoint的唯一资源标识。
user_data	String	用户自定义数据 最大长度支持UTF-8编码后2048字节。
enabled	String	endpoint启用开关 true或false字符串。
token	String	设备token 最大长度512个字节。

状态码： 400

表 9-46 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-47 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-48 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-49 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}/endpoints?  
offset=0&limit=10
```

```
null
```

响应示例

状态码： 200

OK

```
{  
  "request_id": "c90d871f77f7425cae0a8b772a563d17",  
  "next_page_flag": false,  
  "endpoints": [ {  
    "create_time": "2018-01-12T01:25:14Z",  
    "endpoint_urn": "urn:smn:regionId:429ffced18074da0938112f2c362b935:endpoint-APNS-  
application_name-62f0bafec61c45fbb5d0fc6bd696a2d",  
    "user_data": "no user data",  
    "enabled": "true",  
    "token": "1f2fda4c1013c47ac4c59bd22379d88e14b07b0ebf47f2f79d1a6b06ed03390d"  
  } ]  
}
```

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.2.2 创建 Application endpoint (废弃)

功能介绍

创建应用平台的endpoint终端。

URI

POST /v2/{project_id}/notifications/applications/{application_urn}/endpoints

表 9-50 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
application_urn	是	String	Application的唯一资源标识， 可通过查询Application获取该标识。

请求参数

表 9-51 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 9-52 请求 Body 参数

参数	是否必选	参数类型	描述
token	是	String	移动应用设备token，最大长度512个字节。
user_data	是	String	用户自定义数据，最大长度支持UTF-8编码后2048字节。

响应参数

状态码： 201

表 9-53 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
endpoint_urn	String	Endpoint的唯一资源标识。

状态码： 400

表 9-54 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-55 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-56 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-57 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

创建Application endpoint

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}/endpoints
{
  "token": "3708232124742383445",
  "user_data": "This is token for user001"
}
```

响应示例

状态码： 201

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "endpoint_urn": "urn:smn:regionId:f96188c7ccaf4ffb0c9aa149ab2bd57:endpoint-APNS-example_application_name-37a2cb947fa43a20bc66e91281194e7a"
}
```

状态码

状态码	描述
201	OK
400	Bad Request
403	Unauthorized

状态码	描述
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.2.3 查询 Application 的 Endpoint 属性（废弃）

功能介绍

获取endpoint的属性。

URI

GET /v2/{project_id}/notifications/endpoints/{endpoint_urn}

表 9-58 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
endpoint_urn	是	String	Endpoint的唯一资源标识，可通过查询Application的Endpoint列表获取该标识。

请求参数

表 9-59 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-60 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
attributes	attributes object	属性列表。

表 9-61 attributes

参数	参数类型	描述
enabled	String	设备是否可用。
token	String	设备token。
user_data	String	用户数据。

状态码： 400

表 9-62 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-63 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-64 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-65 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

```
GET https://{SMN_Endpoint}/v2/{project_id}/notifications/endpoints/{endpoint_urn}
null
```

响应示例

状态码： 200

OK

```
{
  "request_id" : "c90d871f77f7425cae0a8b772a563d17",
  "attributes" : {
    "enabled" : "true",
    "token" : "3708232124742383445",
    "user_data" : "abc"
  }
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found

状态码	描述
500	Internal Server Error

错误码

请参见[错误码](#)。

9.2.4 更新 Application endpoint (废弃)

功能介绍

更新设备属性。

URI

PUT /v2/{project_id}/notifications/endpoints/{endpoint_urn}

表 9-66 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
endpoint_urn	是	String	Endpoint的唯一资源标识，可通过查询Application的Endpoint列表获取该标识。

请求参数

表 9-67 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 9-68 请求 Body 参数

参数	是否必选	参数类型	描述
enabled	否	String	设备是否可用，值为true或false字符串。
user_data	否	String	用户自定义数据，最大长度支持UTF-8编码后2048字节。

响应参数

状态码： 200

表 9-69 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码： 400

表 9-70 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-71 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-72 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-73 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

更新Application endpoint

```
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/applications/{application_urn}/endpoints
{
  "enabled" : "false",
  "user_data" : "This is an APNS token for user001"
}
```

响应示例

状态码： 200

OK

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

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found

状态码	描述
500	Internal Server Error

错误码

请参见[错误码](#)。

9.2.5 删除 Application endpoint (废弃)

功能介绍

删除设备。

URI

DELETE /v2/{project_id}/notifications/endpoints/{endpoint_urn}

表 9-74 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
endpoint_urn	是	String	Endpoint的唯一资源标识，可通过查询Application的Endpoint列表获取该标识。

请求参数

表 9-75 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 9-76 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

状态码： 400

表 9-77 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-78 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-79 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-80 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

删除Application endpoint

```
DELETE https://{SMN_Endpoint}/v2/{project_id}/notifications/endpoints/{endpoint_urn}
```

```
null
```

响应示例

无

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.3 Application 直发消息操作

9.3.1 App 消息发布（废弃）

功能介绍

将消息直发给endpoint设备。

URI

```
POST /v2/{project_id}/notifications/endpoints/{endpoint_urn}/publish
```

表 9-81 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。 获取项目ID请参考 获取项目ID 。
endpoint_urn	是	String	Endpoint的唯一资源标识，可通过查询Application的Endpoint列表获取该标识

请求参数

表 9-82 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。 通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 9-83 请求 Body 参数

参数	是否必选	参数类型	描述
message	否	String	<p>message与message_structure二者选其一。</p> <p>message, App消息发布。</p> <p>message_structure, 使用消息结构体方式的App消息发布。</p> <p>app推送的消息内容, 当前支持的推送平台有HMS、APNS、APNS_SANDBOX。</p> <p>HMS是为开发者提供的消息推送平台。</p> <p>APNS和APNS_SANDBOX是用于推送iOS消息的服务平台。</p> <p>HMS平台指定的消息内容不超过2K。</p> <p>APNS和APNS_SANDBOX平台的消息内容不能超过4K。</p> <p>推送平台的消息内容格式要求详情见application消息体格式。</p> <p>华为透传消息</p> <pre>{ "hps": { "msg": { "type": 1, "body": { "key": "value" } } }</pre> <p>华为系统通知栏消息</p> <pre>{ "hps": { "msg": { "type": 3, "body": { "content": "Push message content", "title": "Push message content" }, "action": { "type": 1, "param": { "intent": "#Intent;compo=com.rvr/.Activity;S.W=U;end" } } }, "ext": { "biTag": "Trump", "icon": "http://upload.w.org/00/150pxsvg.png" } }</pre> <p>苹果平台消息格式 { "aps": { "alert": "hello world" } }</p>

参数	是否必选	参数类型	描述
message_structure	否	String	<p>app推送的消息内容，当前支持的推送平台有HMS、APNS、APNS_SANDBOX。</p> <p>HMS是为开发者提供的消息推送平台。</p> <p>APNS和APNS_SANDBOX是用于推送iOS消息的服务平台。</p> <p>HMS平台指定的消息内容不超过2K。</p> <p>APNS和APNS_SANDBOX平台的消息内容不能超过4K。</p> <p>推送平台的消息内容格式要求详情见application消息体格式。</p> <p>华为透传消息</p> <pre>{ "HMS": { "hps": { "msg": { "type": 1, "body": { "key": "value" } } } } }</pre> <p>华为系统通知栏消息</p> <pre>{ "HMS": { "hps": { "msg": { "type": 3, "body": { "content": "Push message content", "title": "Push message content" }, "action": { "type": 1, "param": { "intent": "#Intent;compo=com.rvr/.Activity;S.W=U;end" } } }, "ext": { "biTag": "Trump", "icon": "http://upload.w.org/00/150pxsvg.png" } } } }</pre> <p>苹果平台消息格式</p> <pre>{ "APNS": { "aps": { "alert": "hello world" } } }</pre>

参数	是否必选	参数类型	描述
time_to_live	否	String	<p>消息发送的生存时间，是相对于发布时间的。</p> <p>SMN系统将移动推送消息转交给推送平台前，会计算该消息在系统消耗的时间。只有消耗的时间小于time_to_live时，SMN才会将消息转交给推送平台，并将time_to_live减去消耗的时间传递给推送平台，否则消息废弃。</p> <p>time_to_live的单位是s，变量默认值是3600s，即一小时。值为正整数且小于等于3600*24。</p> <p>缺省值：3600</p>

响应参数

状态码： 200

表 9-84 响应 Body 参数

参数	参数类型	描述
message_id	String	唯一的消息ID。
request_id	String	请求的唯一标识ID。

状态码： 400

表 9-85 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 403

表 9-86 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。

参数	参数类型	描述
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 404

表 9-87 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

状态码： 500

表 9-88 响应 Body 参数

参数	参数类型	描述
request_id	String	请求的唯一标识ID。
code	String	服务异常错误信息编码。
message	String	服务异常错误信息描述。

请求示例

- App消息发布
POST https://{SMN_Endpoint}/v2/{project_id}/notifications/endpoints/{endpoint_urn}/publish


```
{
  "message": {
    "hps": {
      "msg": {
        "action": {
          "param": {
            "intent": "#Intent;compo=com.HC.SmnAppPush.defaultAction/.Activity;S.W=U;end"
          },
          "type": 1
        },
        "body": {
          "title": "hello world",
          "content": "hello world"
        },
        "type": 3
      }
    }
  },
  "time_to_live": 3600
}
```

- 使用消息结构体方式的App消息发布

POST https://{SMN_Endpoint}/v2/{project_id}/notifications/endpoints/{endpoint_urn}/publish

```
{
  "message": {
    "HMS": {
      "hps": {
        "msg": {
          "action": {
            "param": {
              "intent": "#Intent;compo=com.HC.SmnAppPush.defaultAction/.Activity;S.W=U;end"
            },
            "type": 1
          },
          "body": {
            "title": "hello world",
            "content": "hello world"
          },
          "type": 3
        }
      }
    }
  },
  "time_to_live": 3600
}
```

响应示例

状态码： 200

OK

```
{
  "request_id": "6a63a18b8bab40ffb71ebd9cb80d0085",
  "message_id": "bf94b63a5dfb475994d3ac34664e24f2"
}
```

状态码

状态码	描述
200	OK
400	Bad Request
403	Unauthorized
404	Not Found
500	Internal Server Error

错误码

请参见[错误码](#)。

9.4 订阅确认 V2 接口（废弃）

用途

订阅确认接口

接口

PUT	/v2/{tenant_id}/notifications/topics/{topic_urn}/subscriptions/{subscription_urn}/confirmation
-----	--

PUT	/v2/{tenant_id}/notifications/subscriptions/confirmation
-----	--

邮件模板生命周期管理接口（废弃）

POST	/v2/{tenant_id}/notifications/integrated_system_message_template
------	--

联系人生命周期管理接口（废弃）

POST	/v2/notifications/contact/callback/{event_type}
------	---

P12证书转Cert证书（废弃）

POST	/rest/v2/notifications/credentials_file_upload
------	--

A 附录

A.1 Topic 属性表

属性名称	属性作用	属性默认值
access_policy	Topic资源的访问策略，控制其他用户访问该Topic的权限。	空字符串。
introduction	在邮件和短信订阅确认信息中展示Topic的介绍，使您能够让订阅者对主题有更多的了解。	空字符串。

A.2 通用请求返回值

A.2.1 异常响应

- 异常响应要素说明

名称	参数类型	说明
request_id	String	请求的ID
code	String	参考 错误码
message	String	参考 错误码

- 异常响应举例

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

A.2.2 状态码

表 A-1 正常返回值

返回值	说明
200	任务提交成功
201	创建成功

表 A-2 异常返回值

返回值	说明
400 Bad Request	请求参数错误。
401 Unauthorized	鉴权失败。
403 Forbidden	没有权限对资源访问。
404 Not Found	请求的资源不存在。
500 Internal Server Error	请求未完成。服务异常。
501 Not Implemented	请求未完成。服务器不支持所请求的功能。
502 Bad Gateway	请求未完成。服务器从上游服务器收到一个无效的响应。
503 Service Unavailable	请求未完成。系统暂时异常。
504 Gateway Timeout	网关超时。

A.3 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

状态码	错误码	错误信息	描述	处理措施
200	SMN.0190	This endpoint already exists.	endpoint已存在。	该endpoint在列表中已经存在。
400	SMN.0002	Parameter: Name is invalid.	参数: name 无效。	请输入有效的topic名称
400	SMN.0003	Parameter: DisplayName is invalid.	参数: displayname 无效。	请输入有效的topic显示名

状态码	错误码	错误信息	描述	处理措施
400	SMN.0005	Parameter: TopicUrn is invalid.	参数: topic_urn无效	请输入有效的 topic_urn参数。
400	SMN.0011	Parameter: Protocol is invalid.	参数: protocol无效。	请输入正确的协议类型
400	SMN.0012	Parameter: Endpoint is invalid.	参数: endpoint无效。	请输入正确格式的终端。
400	SMN.0014	Parameter: SubscriptionUrn is invalid.	参数: SubscriptionUrn 无效。	请输入有效的 subscription_urn。
400	SMN.0015	Parameter: Offset or limit is invalid.	参数: offset or limit无效。	请输入有效的 offset和limit
400	SMN.0017	Parameter: Remark is invalid.	参数: remark无效。	请输入有效的 remark参数
400	SMN.0019	Producer Exceptions.	生成器异常。	如遇到请联系技术支持。
400	SMN.0021	MessageStructure is invalid.	消息结构无效。	请输入有效的 message_structure参数
400	SMN.0024	Parameter: content is invalid.	参数: content无效。	请输入有效的 content参数。
400	SMN.0025	Template already exists.	模板已经存在。	请不要重复创建相同的模板
400	SMN.00290	SubscriberUrn is unavailable.	SubscriberUrn不可用。	检查SubscriberUrn是否正确。
400	SMN.0031	Parameter: template_id is invalid.	参数: template_id无效。	请输入有效的 template_id参数。
400	SMN.0032	Parameter: message_template_name is invalid.	参数: message_template_name无效。	请输入有效的 message_template_name参数。
400	SMN.0033	Parameter: push_policy is invalid.	参数: push_policy无效。	请输入有效的 push_policy参数。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0034	Parameter: subscription_auth_policy is invalid.	参数: subscription_auth_policy 无效。	请输入有效的 subscription_auth_policy 参数。
400	SMN.0035	Parameter: validation_policy is invalid.	参数: validation_policy 无效。	请输入有效的 validation_policy 参数。
400	SMN.0036	Parameter: xdomain_type is invalid.	参数: xdomain_type 无效。	请输入有效的 xdomain_type 参数。
400	SMN.0038	Parameter: tag is invalid.	参数: tag 无效。	请输入有效的 tag 参数。
400	SMN.0042	Parameter: Message_template_name is invalid.	参数: message_template_name 无效。	请输入有效的 message_template_name 参数。
400	SMN.0043	Parameter: subscription is invalid.	参数: subscription 无效。	请输入有效的 subscription 参数。
400	SMN.0044	Exceeded template limit.	模板数量超过限制。	消息模板数量超过限制。每个用户可以创建100个消息模板。
400	SMN.0046	Parameter: Attribute name is invalid.	参数: attribute_name 无效。	输入有效的 attribute_name。
400	SMN.0048	Parameter: Access policy is invalid.	参数: access_policy 无效。	请输入有效的 access_policy 值。
400	SMN.0049	Parameter: Access policy version is invalid.	参数: access_policy 版本无效。	请输入 access_policy 规范的版本。
400	SMN.0050	Parameter: Access policy ID is invalid.	参数: access_policy id 无效。	请输入有效的 access_policy id 值。
400	SMN.0051	Parameter: Access policy statement is invalid.	参数: access_policy statement 无效。	请输入有效的 access_policy statement 值。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0052	Parameter: Access policy statement ID is invalid.	参数: access_policy statement id 无效。	请输入有效的 access_policy statement id 值。
400	SMN.0053	Parameter: Access policy statement resource is invalid.	参数: access_policy statement resource 无效。	请输入有效的 access_policy statement resource 值。
400	SMN.0054	Parameter: Access policy statement effect is invalid.	参数: access_policy statement effect 无效。	请输入有效的 access_policy statement effect 值。
400	SMN.0055	Parameter: Access policy statement action is invalid.	参数: access_policy statement action 无效。	请输入有效的 access_policy statement action 值。
400	SMN.0056	Parameter: Access policy statement principle is invalid.	参数: access_policy statement principle 无效。	请输入有效的 access_policy statement principle 值。
400	SMN.0058	Parameter: Access policy statement condition is invalid.	参数: access_policy statement condition 无效。	请输入有效的 access_policy statement condition 值。
400	SMN.0059	Parameter: job_id is invalid.	参数: job_id 无效。	如遇到请联系技术支持。
400	SMN.0061	Parameter: Service name is invalid.	参数: Service name 无效。	如遇到请联系技术支持。
400	SMN.0062	Parameter: Show name is invalid.	参数: Show name 无效。	如遇到请联系技术支持。
400	SMN.0066	Parameter: System parameter name is invalid.	参数: 系统参数名称无效。	如遇到请联系技术支持。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0067	Parameter: System parameter value is invalid.	参数：系统参 数值无效。	如遇到请联系技术 支持。
400	SMN.0072	Parameter: Service is invalid.	参数：service 无效。	请输入有效的 service参数。
400	SMN.0073	Parameter: Invalid action.	参数:action无 效。	请输入有效的 action参数。
400	SMN.0074	Failed to add permission.	添加权限失 败。	如遇到请联系技术 支持。
400	SMN.0077	Parameter: Status is invalid.	参数：status 无效。	请输入有效的 status参数。
400	SMN.0080	The signId parameter is invalid.	参数：sign_id 无效。	请输入有效的 sign_id参数。
400	SMN.0081	The signName parameter is invalid.	参数： sign_name无 效。	请输入有效的 sign_name参数， 签名只能由中文、 英文字母和数字组 成，长度为2到8个 字符。
400	SMN.0082	The length of the remark parameter value has reached the maximum allowed.	remark参数超 过长度限制。	请输入有效的 remark参数。
400	SMN.0087	The SMS message signature has not been approved.	短信签名未被 审批通过。	请输入已经审批通 过的签名。
400	SMN.0090	Invalid domain ID.	无效的 domain_id。	请输入有效的 domain_id。获取 方法请参见获取项 目ID。
400	SMN.0092	The kafka_id parameter is invalid.	参数： kafka_id无 效。	如遇到请联系技术 支持。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0093	The address_list parameter is invalid.	参数： address_list无效。	如遇到请联系技术支持。
400	SMN.0094	The type of queue parameter is invalid.	队列参数类型无效。	如遇到请联系技术支持。
400	SMN.0095	The count of queue parameter is invalid.	队列参数统计数量无效。	如遇到请联系技术支持。
400	SMN.0097	The names parameter is invalid.	names参数无效。	如遇到请联系技术支持。
400	SMN.0100	The eventType parameter is invalid.	参数： event_type无效。	请输入有效的event_type参数。
400	SMN.0102	Parameter: message_id is invalid.	参数： message_id无效。	请输入有效的message_id参数。
400	SMN.0103	The start_time parameter value is invalid.	start_time参 数值无效。	请输入有效的start_time参数。
400	SMN.0104	The end_time parameter value is invalid.	end_time参数 值无效。	请输入有效的end_time参数。
400	SMN.0108	Parameter: Type is invalid.	参数：type无 效。	请输入有效的type参数。
400	SMN.0110	Parameter: Start date is invalid.	参数： start_date无 效。	请输入有效的start_date参数。
400	SMN.0112	Parameter: End date is invalid.	参数： end_date无 效。	请输入有效的end_date参数。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0113	The start time and end time must be within the past 3 months.	开始时间和结束时间必须在过去3个月内。	开始时间和结束时间必须在过去3个月内。
400	SMN.0114	The end time must be later than the start time.	结束时间必须大于开始时间。	结束时间必须大于开始时间。
400	SMN.0115	Parameter: Application name is invalid.	参数: application name无效。	请输入有效的 application name。
400	SMN.0118	Wrong Certificate.	证书错误。	请输入有效证书。
400	SMN.0121	An application with the same name but different properties already exists.	具有相同名称但属性不同的应用程序已存在。	请输入有效的 application名称
400	SMN.0123	Parameter: ApplicationUrn is invalid.	参数: application_urn无效。	请输入有效的 application_urn参数。
400	SMN.0124	Parameter: application attribute is invalid.	参数: application_attributes无效。	请输入有效的 attributes参数。
400	SMN.0126	ContentType is wrong.	Content类型错误。	如遇到请联系技术支持。
400	SMN.0129	Certificate type is wrong.	证书类型错误。	如遇到请联系技术支持。
400	SMN.0132	The value of the User_data parameter is invalid.	user_data无效。	请输入有效的 user_data。
400	SMN.0133	The device token already exists.	device_token已存在。	device_token不能重复创建

状态码	错误码	错误信息	描述	处理措施
400	SMN.0134	The endpoint URN is invalid.	endpoint_urn无效。	请输入有效的endpoint_urn。
400	SMN.0135	The endpoint does not exist.	endpoint不存在。	请使用存在的endpoint
400	SMN.0136	The value of the enabled parameter is invalid.	enabled无效。	请输入有效的enabled参数。
400	SMN.0137	The application URN is invalid.	application_urn无效。	请输入有效的application_urn参数。
400	SMN.0140	The topic URN configured for the message sending failure event is invalid.	配置发送消息失败回调事件的topic_urn无效。	配置发送消息失败回调事件的topic_urn无效，请输入有效的topic_urn。
400	SMN.0141	The topic configured for the message sending failure event does not exist.	配置发送消息失败回调事件的topic_urn不存在。	配置发送消息失败回调事件的topic_urn不存在，请输入有效的topic_urn。
400	SMN.0142	The topic URN configured for the message sending success event is invalid.	配置发送消息成功回调事件的topic_urn无效。	配置发送消息成功回调事件的topic_urn无效，请输入有效的topic_urn。
400	SMN.0143	The topic configured for the message sending success event does not exist.	配置发送消息成功回调事件的topic_urn不存在。	配置发送消息成功回调事件的topic_urn不存在，请输入有效的topic_urn。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0144	The topic URN configured for the response message event is invalid.	配置回复消息回调事件的topic_urn无效。	配置回复消息回调事件的topic_urn无效，请输入有效的topic_urn。
400	SMN.0145	The topic configured for the response message event does not exist.	配置回复消息回调事件的topic_urn不存在。	配置回复消息回调事件的topic_urn不存在，请输入有效的topic_urn。
400	SMN.0148	Parameter: user_id is invalid.	参数: user_id 无效。	请输入有效的user_id参数。
400	SMN.0150	Complete real-name registration using your company's certificate.	使用贵公司的证书完成实名注册。	如遇到请联系技术支持。
400	SMN.0151	The message signature type cannot be promotional.	不能是推广类消息签名。	请输入通知验证码类短信签名。
400	SMN.0152	The number allocated to the SMS template has reached the maximum allowed.	分配的短信模板数量已达限额。	创建的短信模板超过限额，每个用户可以创建100个短信模板。
400	SMN.0155	Parameter: SmsTemplate Type is invalid.	参数: sms_template_type无效。	请输入有效的sms_template_type，请参见创建短信模板。
400	SMN.0156	Parameter: SmsTemplate Name is invalid.	参数: sms_template_name无效。	请输入有效的sms_template_name，请参见创建短信模板。
400	SMN.0157	Parameter: SmsTemplate Content is invalid.	参数: sms_template_content无效。	请输入有效的sms_template_content，请参见创建短信模板。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0159	The number of endpoints has reached the maximum allowed.	终端数量已达配额。	发送号码个数超过限额，手机号码个数不能超过1000。
400	SMN.0160	Invalid endpoint.	endpoint无效。	请输入有效的endpoint。
400	SMN.0161	Invalid SMS template ID.	短信模板ID无效。	请输入有效的sms_template_id。
400	SMN.0188	Invalid SMS template type.	短信模板类型无效。	当前的短信模板类型不存在，必须是推广模板。
400	SMN.0189	Failed to obtain the certificate required for reviewing the SMS message template function.	获取查看sms消息模板功能所需的证书失败。	如遇到请联系技术支持。
400	SMN.0191	Parameter: Invalid enterprise name.	公司名称无效。	如遇到请联系技术支持。
400	SMN.0192	Parameter: Invalid enterprise certificate ID.	公司证书无效。	如遇到请联系技术支持。
400	SMN.0204	Promotional SMS messages must contain the way to unsubscribe in the end, for example, by replying "TD", or "T".	推广短信末尾必须包含取消订阅的方式，例如，通过回复"TD"或"T"。	在推广短信内容末尾必须包含退订方式，比如在内容末尾添加“退订回TD”。

状态码	错误码	错误信息	描述	处理措施
400	SMN.0215	The SMS message signature ID does not match the signature name in the SMS message template content.	sms 消息签名 id 与 sms 消息模板内容中的签名名称不匹配。	请传入与 sms_template_content 内容中的签名名称一致的签名 ID。
400	SMN.0216	The template content does not include any valid SMS message signature.	模板内容不包括任何有效的 sms 消息签名。	请在内容填入有效的签名。
400	SMN.0217	The template content does not include any SMS message signature.	模板内容不包括任何 sms 消息签名。	sms_template_content 内容必须要包含有效的签名名称，如【XX商城】。
400	SMN.0224	The SMS message template is invalid because the associated signature has been deleted.	短信模板因关联签名删除已失效。	请传入有效的 sms_template_id。
400	SMN.0227	The message contains sensitive words.	消息中包含敏感关键词。	当前传入的 message 中包含敏感关键词，请根据返回提示修改 message 内容。
400	SMN.0230	Parameter: Invalid title.	参数 title 无效。	请输入合法的 title 参数
400	SMN.0261	Check whether your account balance is sufficient and top up your account if necessary.	请检查账户余额是否充足，如余额不足请尽快充值。	请检查账户余额是否充足，如余额不足请尽快充值。

状态码	错误码	错误信息	描述	处理措施
403	SMN.0001	No permission to request resources.	没有请求资源的权限。	添加相应的权限
403	SMN.0004	Exceeded topic limit.	主题数量超过限制。	topic数量超过了限制，每个用户可以最多创建3000个主题。
403	SMN.0007	Exceeded subscription limit.	订阅数量超过限制。	每个主题最多支持10000个订阅者。
403	SMN.0008	Parameter: Subject is invalid.	参数: subject 无效。	请输入有效的subject参数。
403	SMN.0009	Parameter: Message is invalid.	参数: message 无效。	请输入有效的message参数
403	SMN.0022	Parameter: token is invalid.	参数: token 无效。	请输入有效token。
403	SMN.00308	Policy doesn't allow {} to be performed.	策略不允许执行。	请检查SMN权限。
403	SMN.0047	Parameter: Value exceeds the maximum length.	参数: value 超出最大长度。	请输入有效的value值
403	SMN.0057	The number of SMS messages exceeds the limit.	SMS消息的数量超过了限制。	短信直发时，同一个手机号码限制1小时内最多发送60条短信，12小时最多发200条短信。
403	SMN.0065	The number of email messages exceeds the limit.	邮件的数量超过了限制。	如遇到请联系技术支持。
403	SMN.0069	Not authorized to subscribe internal endpoints.	订阅内部终端未授权。	检查http(s)终端地址，请填写正确的https(s)终端地址。

状态码	错误码	错误信息	描述	处理措施
403	SMN.0070	No permission to request resources. The role is op_restricted. Contact customer service.	没有请求资源的权限。操作受限。	请检查余额是否充足，余额不足请尽快充值。如未解决，请联系客服技术支持。
403	SMN.0071	No permission to request resources.Role is op_suspended .	没有请求资源的权限。角色被挂起。	请检查账号是否被挂起冻结。
403	SMN.0075	Parameter: tags are too many.	参数： 标签数过多。	超过消息模板的最大非重复标签数，模板内容中允许最多90个不重复的标签。
403	SMN.0078	The maximum number of subscriptions has been reached.	已达到订阅的最大数量。	订阅者列表超过数量限制。
403	SMN.0083	Finish real-name authentication before continuing.	请先完成实名认证。	如遇到请联系技术支持。
403	SMN.0086	The number allocated to the SMS message signature has reached the maximum allowed.	分配给 sms 消息签名的数字已达限额。	创建签名超过限额，您最多可以创建10个短信签名。

状态码	错误码	错误信息	描述	处理措施
403	SMN.0099	The SMN service has not been authorized to send messages to this queue.	SMN服务未被授权将消息发送到此队列。	请在该队列策略里面授权smn服务相关权限。
403	SMN.0116	Platform do not support yet.	平台暂不支持。	该平台暂不支持。
403	SMN.0117	Certificate type is not supported.	证书类型暂不支持。	该证书类型暂不支持。
403	SMN.0119	PlatformPrinciple must be at most 8192 bytes long in UTF-8 encoding.	UTF-8编码的platform_principle不得超过8192bytes。	请输入有效platform_principle参数，不超过8192bytes，且为UTF-8编码。
403	SMN.0120	PlatformCredential must be at most 8192 bytes long in UTF-8 encoding.	platform_credential不超过8192bytes，且为UTF-8编码。	请输入有效platform_credential参数，不超过8192bytes，且为UTF-8编码。
403	SMN.0127	File is too large.	文件过大。	如遇到请联系技术支持。
403	SMN.0128	Password is empty.	密码为空。	如遇到请联系技术支持。
403	SMN.0131	The value of the Token parameter is invalid.	token无效。	请输入有效的token。
403	SMN.0147	Only enterprise users who use the certificate images for authentication can create message signatures.	只有使用证书进行身份验证的企业用户才能创建消息签名。	如遇到请联系技术支持。

状态码	错误码	错误信息	描述	处理措施
403	SMN.0153	Only enterprise users can create SMS message Template.	只有企业用户才能创建短信模板。	请完成企业身份认证。
403	SMN.0154	Only enterprise users who use the certificate images for authentication can create sms message templates.	只有使用证书进行身份验证的企业用户才能创建短信模板。	如遇到请联系技术支持。
403	SMN.0165	Parameter: tenant_token is invalid.	参数:tenant_token无效。	请输入有效的tenant_token参数。
403	SMN.0195	Complete enterprise real-name authentication first.	请先完成实名认证。	请进行实名认证。
403	SMN.0228	Do not have permission to use this API. Contact customer service to apply for permission.	当前接口没有使用权限，如需使用请联系客服申请。	当前接口目前没有权限访问，如需使用请联系技术支持申请。
404	SMN.0006	Topic not found.	未查询到该主题。	topic不存在。
404	SMN.0013	Subscription resource not found.	查询不到订阅资源。	请使用正确的订阅资源
404	SMN.0027	Template not found.	找不到模板。	请使用存在的模板。
404	SMN.0045	Attribute not found.	未查询到attribute。	请使用存在的属性。
404	SMN.0060	Job not found.	查询不到Job。	如遇到请联系技术支持。

状态码	错误码	错误信息	描述	处理措施
404	SMN.0063	Cloud service not found.	找不到云服务。	如遇到请联系技术支持。
404	SMN.0068	System parameter not found.	找不到系统参数。	如遇到请联系技术支持。
404	SMN.0076	Default message template not found.	默认消息模板不存在。	请先创建一个 default 模板。
404	SMN.0084	The SMS message signature could not be found.	找不到SMS短信签名。	请输入已经审批通过的签名。
404	SMN.0091	Default system message template not found.	不存在默认系统消息模板。	请先创建一个默认模板。
404	SMN.0096	Kafka not found.	找不到Kafka。	如遇到请联系技术支持。
404	SMN.0101	The graph does not exist in the functiongraph service.	函数在functiongraph服务中不存在。	请检查functiongraph终端是否输入正确。
404	SMN.0109	Start date is not found.	找不到参数start_date。	请输入有效的start_date参数，不能为空。
404	SMN.0111	End date is not found.	找不到参数end_date。	请输入有效的end_date参数，不能为空。
404	SMN.0122	Application not found.	找不到应用程序。	请输入有效的application名称
404	SMN.0125	Endpoint does not exist.	终端不存在。	请输入存在的application endpoint
404	SMN.0130	The application does not exist.	平台不存在。	请输入有效的application_urn。

状态码	错误码	错误信息	描述	处理措施
404	SMN.0162	SMS template not found.	短信模板不存在。	短信模板不存在，请输入有效的短信模板。
404	SMN.0167	Phone number does not exist.	手机号码不存在。	用户注册的手机号码不存在。
404	SMN.0168	Email Address does not exist.	邮件地址不存在。	用户注册的邮件地址不存在。
429	SMN.00359	Topic publishing exceeds the flow control limit, please try again later.	主题发布超过流控限制。	请稍后重试。
429	SMN.00362	Topic publishing exceeds the global flow control limit, please try again later.	主题发布超过全局流控限制。	请稍后重试。
500	SMN.0016	Database Exceptions.	数据库异常。	如遇到请联系技术支持。
500	SMN.0018	Service internal error.	服务内部错误。	如遇到请联系技术支持。
500	SMN.0089	An error occurred in the BSS system. Try again later.	BSS系统中出现错误。请稍后再试。	如遇到请联系技术支持。
500	SMN.0158	Failed to upload the file.	上传文件失败。	如遇到请联系技术支持。

A.4 国家码和地区码

国家（或地区）	中文名	国家（或地区）码
Afghanistan	阿富汗	93
Albania	阿尔巴尼亚	355

国家（或地区）	中文名	国家（或地区）码
Algeria	阿尔及利亚	213
American Samoa	美属萨摩亚	1684
Andorra	安道尔	376
Angola	安哥拉	244
Anguilla	安圭拉	1264
Antigua and Barbuda	安提瓜和巴布达	1268
Argentina	阿根廷	54
Armenia	亚美尼亚	374
Aruba	阿鲁巴	297
Australia	澳大利亚	61
Austria	奥地利	43
Azerbaijan	阿塞拜疆	994
Bahamas	巴哈马	1242
Bahrain	巴林	973
Bangladesh	孟加拉国	880
Barbados	巴巴多斯	1246
Belarus	白俄罗斯	375
Belgium	比利时	32
Belize	伯利兹	501
Benin	贝宁	229
Bermuda	百慕大群岛	1441
Bhutan	不丹	975
Bolivia	玻利维亚	591
Bosnia and Herzegovina	波斯尼亚和黑塞哥维那	387
Botswana	博茨瓦纳	267
Brazil	巴西	55
British Indian Ocean Territory	英属印度洋领地	246
Brunei	文莱	673

国家（或地区）	中文名	国家（或地区）码
Bulgaria	保加利亚	359
Burkina Faso	布基纳法索	226
Burundi	布隆迪	257
Cambodia	柬埔寨	855
Cameroon	喀麦隆	237
Canada	加拿大	1
Cape Verde	佛得角	238
Cayman Islands	开曼群岛	1345
Central African Republic	中非共和国	236
Chad	乍得	235
Chile	智利	56
Colombia	哥伦比亚	57
Comoros	科摩罗	269
Republic Of The Congo	刚果共和国	242
Democratic Republic of the Congo	刚果民主共和国	243
Cook Islands	库克群岛	682
Costa Rica	哥斯达黎加	506
Croatia	克罗地亚	385
Curacao	库拉索	599
Cyprus	塞浦路斯	357
Czechia	捷克	420
Denmark	丹麦	45
Djibouti	吉布提	253
Dominica	多米尼克	1767
Dominican Republic	多米尼加共和国	1809
Ecuador	厄瓜多尔	593
Egypt	埃及	20

国家（或地区）	中文名	国家（或地区）码
El Salvador	萨尔瓦多	503
Equatorial Guinea	赤道几内亚	240
Eritrea	厄立特里亚	291
Estonia	爱沙尼亚	372
Eswatini	斯威士兰	268
Ethiopia	埃塞俄比亚	251
Falkland Islands	福克兰群岛	500
Faroe Islands	法罗群岛	298
Fiji	斐济	679
Finland	芬兰	358
France	法国	33
French Guiana	法属圭亚那	594
French Polynesia	法属波利尼西亚	689
Gabon	加蓬	241
Gambia	冈比亚	220
Georgia	格鲁吉亚	995
Germany	德国	49
Ghana	加纳	233
Gibraltar	直布罗陀	350
Greece	希腊	30
Greenland	格陵兰岛	299
Grenada	格林纳达	1473
Guadeloupe	瓜德罗普岛	590
Guatemala	瓜地马拉	502
Guernsey	根西	44
Guinea	几内亚	224
Guinea-Bissau	几内亚比绍	245
Guyana	圭亚那	592
Haiti	海地	509
Honduras	洪都拉斯	504

国家（或地区）	中文名	国家（或地区）码
Hong Kong (China)	中国香港	852
Hungary	匈牙利	36
Iceland	冰岛	354
India	印度	91
Indonesia	印度尼西亚	62
Iraq	伊拉克	964
Ireland	爱尔兰	353
Isle of Man	马恩岛	44
Israel	以色列	972
Italy	意大利	39
Ivory Coast	科特迪瓦	225
Jamaica	牙买加	1876
Japan	日本	81
Jersey	泽西岛	44
Jordan	约旦	962
Kazakhstan	哈萨克斯坦	7
Kenya	肯尼亚	254
Kuwait	科威特	965
Kyrgyzstan	吉尔吉斯斯坦	996
Laos	老挝	856
Latvia	拉脱维亚	371
Lebanon	黎巴嫩	961
Lesotho	莱索托	266
Liberia	利比里亚	231
Libya	利比亚	218
Liechtenstein	列支敦士登	423
Lithuania	立陶宛	370
Luxembourg	卢森堡	352
Macao (China)	中国澳门	853
Madagascar	马达加斯加	261

国家（或地区）	中文名	国家（或地区）码
Malawi	马拉维	265
Malaysia	马来西亚	60
Maldives	马尔代夫	960
Mali	马里	223
Malta	马耳他	356
Marshall Islands	马绍尔群岛	692
Martinique	马提尼克	596
Mauritania	毛里塔尼亚	222
Mauritius	毛里求斯	230
Mexico	墨西哥	52
Moldova	摩尔多瓦	373
Monaco	摩纳哥	377
Mongolia	蒙古	976
Montenegro	黑山	382
Montserrat	蒙特塞拉特岛	1664
Morocco	摩洛哥	212
Mozambique	莫桑比克	258
Myanmar	缅甸	95
Namibia	纳米比亚	264
Nauru	瑙鲁	674
Nepal	尼泊尔	977
Netherlands	荷兰	31
New Caledonia	新喀里多尼亚	687
New Zealand	新西兰	64
Nicaragua	尼加拉瓜	505
Niger	尼日尔	227
Nigeria	尼日利亚	234
Niue	纽埃岛	683
Norfolk Island	诺福克岛	672
North Macedonia	马其顿	389

国家（或地区）	中文名	国家（或地区）码
Norway	挪威	47
Oman	阿曼	968
Pakistan	巴基斯坦	92
Palau	帕劳	680
Palestinian Territory	巴勒斯坦	970
Panama	巴拿马	507
Papua New Guinea	巴布亚新几内亚	675
Paraguay	巴拉圭	595
Peru	秘鲁	51
Philippines	菲律宾	63
Poland	波兰	48
Portugal	葡萄牙	351
Qatar	卡塔尔	974
Réunion Island	留尼汪	262
Romania	罗马尼亚	40
Rwanda	卢旺达	250
Saint Kitts and Nevis	圣基茨和尼维斯	1869
Saint Lucia	圣卢西亚	1758
Saint Pierre and Miquelon	圣皮埃尔和密克隆	508
Saint Vincent and The Grenadines	圣文森特和格林纳丁斯	1784
Samoa	萨摩亚	685
San Marino	圣马力诺	378
Sao Tome and Principe	圣多美和普林西比	239
Saudi Arabia	沙特阿拉伯	966
Senegal	塞内加尔	221
Serbia	塞尔维亚	381
Seychelles	塞舌尔	248

国家（或地区）	中文名	国家（或地区）码
Sierra Leone	塞拉利昂	232
Singapore	新加坡	65
Sint Maarten (Dutch Part)	荷属圣马丁	1721
Slovakia	斯洛伐克	421
Slovenia	斯洛文尼亚	386
Solomon Islands	所罗门群岛	677
Somalia	索马里	252
South Africa	南非	27
South Korea	韩国	82
Spain	西班牙	34
Sri Lanka	斯里兰卡	94
Suriname	苏里南	597
Sweden	瑞典	46
Switzerland	瑞士	41
Taiwan (China)	中国台湾	886
Tajikistan	塔吉克斯坦	992
Tanzania	坦桑尼亚	255
Thailand	泰国	66
Timor L'este	东帝汶	670
Togo	多哥	228
Tonga	汤加	676
Trinidad and Tobago	特立尼达和多巴哥	1868
Tunisia	突尼斯	216
Türkiye	土耳其	90
Turkmenistan	土库曼斯坦	993
Turks and Caicos Islands	特克斯和凯科斯群岛	1649
Tuvalu	图瓦卢	688
Uganda	乌干达	256

国家（或地区）	中文名	国家（或地区）码
Ukraine	乌克兰	380
United Arab Emirates	阿拉伯联合酋长国	971
United Kingdom	英国	44
United States	美国	1
Uruguay	乌拉圭	598
Uzbekistan	乌兹别克斯坦	998
Vanuatu	瓦努阿图	678
Venezuela	委内瑞拉	58
Vietnam	越南	84
Virgin Islands, British	英属维尔京群岛	1284
Wallis and Futuna	瓦利斯和富图纳	681
Yemen	也门	967
Zambia	赞比亚	260
Zimbabwe	津巴布韦	263