

消息服务

用户指南

用户指南

概述

阿里云消息服务(Message Service)是一种高效、可靠、安全、便捷、可弹性扩展的分布式消息服务。MNS能够帮助应用开发者在他们应用的分布式组件上自由的传递数据，构建松耦合系统。

本引导将向您完整介绍队列模型以及主题订阅模型的使用方法；与访问控制、云监控、日志服务等其他阿里云服务结合使用；同时还提供了各种特色功能介绍以及丰富的最佳实践案例。

资源	描述
阿里云MNS快速入门	本文档将向您介绍十分钟了解阿里云MNS的入门任务
阿里云MNS开发人员指南（本文档）	本文档将为您讲解阿里云MNS服务的核心概念、所有功能介绍与操作步骤，以及与访问控制、监控、日志管理等运维方法
阿里云MNS控制台使用帮助	阿里云MNS管理控制台可以让您通过界面执行MNS的部分功能，本文档为您介绍基于阿里云MNS管理控制台的所有操作
阿里云MNS最佳实践	详细介绍了阿里云MNS的各种使用场景与实践案例
阿里云MNS购买指南	详细介绍了阿里云MNS的价格与计量计费方式
阿里云MNS API手册	详细介绍了阿里云MNS支持的RESTful API操作以及相关的示例
阿里云MNS SDK手册	介绍了官方SDK的调用操作以及示例代码
阿里云MNS开发者工具	列出了一些开发过程中可能使用到的有用工具
阿里云MNS开发者论坛	由开发人员组成的社区形式的论坛，您可以在这里讨论和学习与阿里云MNS有关的技术问题

队列模型

队列操作

- 操作 Queue 的接口

接口	功能描述	支持方式
CreateQueue	创建队列	控制台、API调用
SetQueueAttributes	设置队列属性	控制台、API调用
GetQueueAttributes	获取队列属性	控制台、API调用
DeleteQueue	删除队列	控制台、API调用
ListQueue	获取队列列表	控制台、API调用

- 操作 Message 的接口

接口	功能描述	支持方式
SendMessage	发送消息	控制台、API调用
BatchSendMessage	批量发送消息	API调用
ReceiveMessage	消费消息	控制台、API调用
BatchReceiveMessage	批量消费消息	API调用
DeleteMessage	删除消息	控制台、API调用
BatchDeleteMessage	批量删除消息	API调用
PeekMessage	查看消息	API调用
BatchPeekMessage	批量查看消息	API调用
ChangeMessageVisibility	修改消息下次可消费时间	API调用

接口具体使用方法请参考 [控制台使用帮助](#) 和 [API使用手册](#)。

主题模型

主题操作

- 操作 Topic 的接口

接口	功能描述	支持方式
CreateTopic	创建主题	控制台、API调用
SetTopicAttributes	设置主题属性	控制台、API调用
GetTopicAttributes	获取主题属性	控制台、API调用
DeleteTopic	删除主题	控制台、API调用
ListTopic	获取主题列表	控制台、API调用

- 操作 Subscription 的接口

接口	功能描述	支持方式
Subscribe	创建订阅	控制台、API调用
SetSubscriptionAttributes	设置订阅属性	控制台、API调用
GetSubscriptionAttributes	获取订阅属性	控制台、API调用
Unsubscribe	删除订阅	控制台、API调用
ListSubscriptionByTopic	获取主题的订阅列表	控制台、API调用

- 操作 Message 的接口

接口	功能描述	支持方式
PublishMessage	发布消息	控制台、API调用

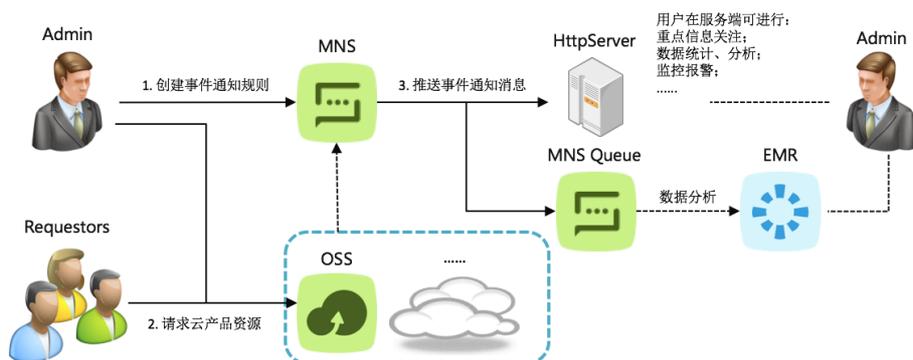
接口具体使用方法请参考 [控制台使用帮助](#) 和 [API使用手册](#)。

事件通知

OSS 事件通知

MNS 推出重磅功能，支持阿里云各大服务通过MNS实现事件通知（回调）。用户可以通过在MNS上创建事件通知的规则，来关注其他阿里云产品指定资源上产生的事件，并且由MNS以消息的方式主动推送到指定的接收端。

- 事件通知按区域提供服务
- 事件通知为异步通知，不会影响用户使用其他云产品的体验
- 事件通知消息在有效时间内不会丢失，当用户接收消息失败时，会按照指定的重试策略重试
- 事件通知消息的接收端目前支持HttpServer和Queue（可以将消息转移到Queue中，由用户来主动消费）
- 事件通知中消息推送的相关介绍请参考主题模型



注意事项

- 事件通知规则创建完，约10分钟后生效
- 在没有其他规则的情况下，默认规则匹配所有资源上的所有事件
- 在有其他规则并且也有默认规则的情况下，不匹配其他规则的事件会匹配默认规则
- 事件通知规则只能创建和删除，目前不支持更新

作用

- 如果您授权其他人可以对您的资源（Bucket）进行某些操作，而您又希望关注这些操作的话，您就可以创建对应的规则，使得这些操作发生时，能够主动通知到您。
- 如果您需要对操作进行统计的话，也可以通过事件通知的方式进行数据分析，查看一段时间内某个操作发生的次数，据此来分析您的业务最近有什么样的发展趋势。并且，您也可以将这些事件通知的消息继续导入其他服务（比如：EMR，进一步进行您自定义的分析）
- 您也可以通过事件通知监控您的资源上发生的某些操作的QPS等等指标。

OSS事件通知

OSS 支持将相关操作产生的事件通知到指定的接收端。其中：

1) 不会触发事件通知

- 失败的操作目前不会触发事件通知
- 同步产生的操作不会触发事件通知

2) 会触发事件通知：

- Lifecycle产生的操作会触发事件通知
- 镜像回写会触发两个事件：ObjectDownloaded:GetObject（由于文件还未拉取，因此etag等信息为空）和ObjectCreated:PutObject。

3) 自定义参数

- OSS 事件通知还支持用户自定义参数，会在消息中将用户的合法自定义参数放在xVars字段中（自定义参数方法与OSS的Callback保持一致，可参考：[这里](#)）。

1. OSS请求返回值

在请求OSS进行相关操作时，事件通知不会影响原定接口的返回值，是通过在返回的Response中增加Header来描述触发事件通知的结果的。如果没有匹配事件通知的规则，则不会增加Header；如果匹配到了，不管事件通知是否触发成功，都会增加这个Header。Header格式如下：

- key：x-oss-process-status
- value：经过Base64编码，解码后为：

```
{
  "code": "Success", //此次触发事件通知的结果，如果成功，则为Success，否则为Fail，需要注意的是，Success只表示oss将消息发送到了topic中，并不代表topic将消息推送到了endpoint那里
  "message": "NotificationSucceed", //此次操作的详细描述，如果失败，此处会说明失败的原因
  "type": "EventNotification", //操作类型，事件通知统一为"EventNotification"，判断的时候需要先判断此类型
  "version": "1.0"} //版本号
```

示例：

```
x-oss-process-status:
'ewogICAgImNvZGUhOiAiU3VjY2VzcyIsCiAgICAibWVzc2FnZSI6ICJOb3RpZmljYXRpb25TdWNjZWVkiiwKICAgICJ0eXB
IjogIkV2ZW50Tm90aWZpY2F0aW9'
```

2. 举例说明如何使用OSS事件通知

某个用户创建了一个Bucket：event-notification-test，然后他关心有哪些Object被上传到这个Bucket中了，于是他就可以创建下面这样的事件通知规则：

- 规则名称：event-notification-test-rule
- 资源描述：event-notification-test/
- 事件类型：PutObject、PostObject
- 接收终端：event-notification-test-queue

以此为例，[五步玩转OSS事件通知](#)，请参考：[这里](#)

默认规则

如果用户简单的想关注自己所有Bucket上产生的所有事件，可以简单的创建一条默认规则就可以了，即，[默认事件通知](#)，请参考：[这里](#)

注意：

- 如果仅有默认规则，则默认规则会匹配所有Bucket上产生的所有事件，但一旦创建其他规则后，默认规则的语义就变成了：[不匹配其他规则的情况下产生的事件通知才会匹配默认规则](#)

3. OSS事件类型

OSS支持的事件类型包括：

注：[请求失败时](#)，目前是[不会触发事件通知的](#)。

事件名称	说明
ObjectCreated:PutObject	
ObjectCreated:PostObject	
ObjectCreated:CopyObject	
ObjectCreated:InitiateMultipartUpload	
ObjectCreated:UploadPart	
ObjectCreated:UploadPartCopy	
ObjectCreated:CompleteMultipartUpload	
ObjectCreate:AppendObject	
ObjectDownload:GetObject	
ObjectRemove>DeleteObject	
ObjectRemove>DeleteObjects	
ObjectRemoved:AbortMultipartUpload	

4. OSS事件通知消息格式

- OSS的事件通知消息内容是经过Base64编码的，解码后是Json格式，具体内容如下：

```
{
  "events": [
    {
      "eventName": "", //事件通知类型
      "eventSource": "", //消息源，固定为"acs:oss"
      "eventTime": "", //事件时间，格式为ISO-8601
      "eventVersion": "", //版本号，目前为"1.0"
      "oss": {
```

```

"bucket": {
  "arn": "", //bucket的唯一标识符, 格式为"acs:oss:region:uid:bucket"
  "name": "", //bucket名称
  "ownerIdentity": ""}, //bucket的owner
  "object": {
    "deltaSize":, //object大小的变化量, 比如新增一个文件, 这个值就是文件大小, 如果是覆盖一个文件, 这个值就是新文件与旧文件的差值, 因此可能为负数
    "eTag": "", //object的etag, 与GetObject()请求返回的ETag头的内容相同
    "key": "", //object名称
    "position":, //可选项, 只有在ObjectCreated:AppendObject事件中才有, 表示此次请求开始append的位置, 注意是从0开始
    "readFrom":, //可选项, 只有在ObjectDownloaded:GetObject事件中才有, 表示文件开始读取的位置, 如果不是Range请求, 则此项为0, 否则则是Range请求的开始字节, 注意是从0开始
    "readTo":, //可选项, 只有在ObjectDownloaded:GetObject事件中才有, 表示文件最后读取的位置, 如果不是Range请求, 则此项为文件的大小, 否则则是Range请求的结束字节增1
    "size": }, //object大小
    "ossSchemaVersion": "", //此字段域的版本号, 目前为"1.0"
    "ruleId": "GetObject"}, //此事件匹配的规则ID
    "region": "", //bucket所在的region
    "requestParameters": {
      "sourceIPAddress": ""}, //请求的源IP
      "responseElements": {
        "requestId": ""}, //请求对应的requestid
        "userIdentity": {
          "principalId": ""}, //请求发起者的uid
          "xVars": { //oss的callback功能中的自定义参数
            "x:callback-var1":"value1",
            "x:callback-var2":"value2"}}}}

```

示例：

```

{"events": [{
  "eventName": "ObjectDownloaded:GetObject",
  "eventSource": "acs:oss",
  "eventTime": "2016-07-01T11:17:30.000Z",
  "eventVersion": "1.0",
  "oss": {
    "bucket": {
      "arn": "acs:oss:cn-shenzhen:1148930107246818:event-notification-test-shenzhen",
      "name": "event-notification-test-shenzhen",
      "ownerIdentity": "1148930107246818"},
      "object": {
        "deltaSize": 0,
        "eTag": "0CC175B9C0F1B6A831C399E269772661",
        "key": "test",
        "readFrom": 0,
        "readTo": 1,
        "size": 1},
        "ossSchemaVersion": "1.0",
        "ruleId": "GetObjectRule"},
        "region": "cn-shenzhen",
        "requestParameters": {
          "sourceIPAddress": "140.205.128.90"},
          "responseElements": {
            "requestId": "5776514AF09A9E6542425D2B"},

```

```
"userIdentity": {
  "principalId": "1148930107246818"},
  "xVars": {
    "x:callback-var1": "value1",
    "x:vallback-var2": "value2"}}}
```

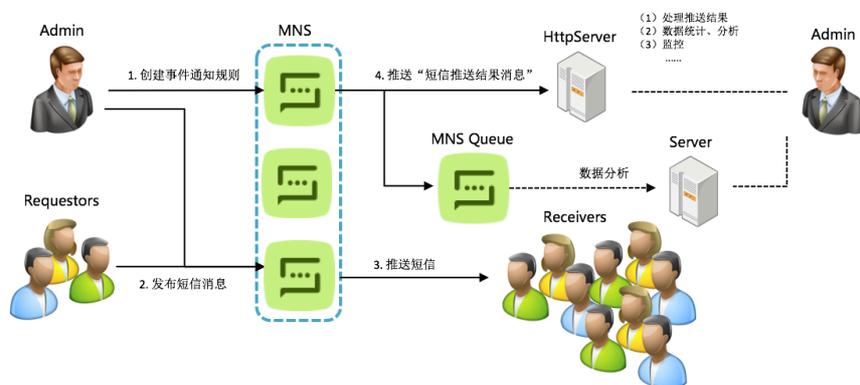
短信推送事件通知

MNS短信推送提供了事件通知功能，支持将短信推送的结果以消息的方式异步通知给用户。用户可以通过创建事件通知的规则，来指定在短信推送过程中触发不同的事件时，将短信推送的结果消息推送到对应的接收端中。

可以解决的问题：

- 异步获取短信推送的结果，再根据业务需求进行相应的处理，比如重新发送
- 对于大批量的短信通知场景，可异步统计推送情况
- 可以自动处理结果消息，对推送情况进行监控，比如发生突发的频繁消息推送，可以第一时间知晓
- 结果消息可以通过多种方式推送到不同的接收端（可能有多个业务部门均关心该数据），以进行不同的统计处理
- 如果A将发送短信的权限授予B，A可以通过该方式来知晓B发送短信的情况。

示意图：



（一）事件通知规则组成

事件通知规则由两部分组成：

- 事件类型
- 消息接收终端

（1）事件类型事件通知是按区域提供服务的，在创建规则时，关注某个（些）事件就是指关心这个区域的短信

推送过程中产生的所有该类型事件。短信推送的事件分为如下三种：

事件类型	描述
SMSRequest:SendSuccessfully	短信推送成功
SMSRequest:SendFailed	短信推送失败
SMSExtend:ReplyMessage	短信推送回复消息

(2) 消息接收终端创建规则时，消息的接收终端可以有两种设置方式：

- 指定接收消息的Http服务或者MNS的队列：此时系统会自动帮忙创建一个主题，并且以指定的Http服务或者MNS队列作为Endpoint，创建相应的订阅。（如果实现没有主题，建议使用这种方式）
- 直接指定主题：此时，规则匹配时，消息就会发布到主题中（此时用户可以选择已有的主题，或者先创建主题，再选择某个主题。**如果用户指定了不存在的主题，消息是没办法正常推送的**）。然后用户可以在该主题上创建订阅，指定通过何种方式将消息推送到指定的接收端。注：
 - **不要选择短信专用的主题（ sms.xxx ）作为消息接收端，短信专用Topic仅支持短信推送。**
 - **如果创建规则时，直接选择主题，只能选择一个。**

注：

事件通知规则指定消息接收端时，可以指定多个。需要注意的是，**选择“主题”，与选择其他的接收终端是互斥的**。即，如果选择了“主题”，那只能选择这一个主题；如果选择了“非主题”的终端，那只能选择非主题的终端。

补充：

- 消息通知是基于MNS的主题实现的，支持多种推送方式，具体可参考：[here](#)。
- 接收消息的Http服务搭建可参考：[说明](#)，[SDK示例](#)。
- 从对应的队列中获取通知消息的方式可参考：[here](#)。

（二）短信推送异步通知消息

短信推送的结果消息如下：

(1) 发送成功：

包括如下字段：

字段名	描述
messageID	消息编号
receiver	接收短信的手机号
state	发送状态（1成功，2失败）
biz_id	编号（可忽略）
template_code	模板code
sms_count	短信条数

receive_time	短信接收的时间
event	触发的事件类型
ver	版本号

```
messageID=$messageID&receiver=$number&state=$state&biz_id=$id&template_code=$code&sms_count=$count&receive_time=$time&ver=$ver&event=$event
```

示例：

```
messageID=23B500FASF59CA1B-1-15B5567FC70-200000009&receiver=12345678901&state=1&biz_id=103245234562^1324561234567&template_code=SMS_12346789&sms_count=1&receive_time=2017-03-31 11:28:19&ver=1.0&event=SendSuccessfully
```

(2) 发送失败

包括如下字段：

字段名	描述
messageID	消息编号
receiver	短信接收号码
state	发送状态（1成功，2失败）
err_code	错误码
event	触发的事件类型

注：err_code包括了两部分：“请求报错”和“运营商给的报错”：

- 请求错误信息请参考：[here](#)。
- 运行商错误信息请参考：[here](#)。

(2.1) 发送多个手机号时，模板或签名错，分别返回多个异步信息，包含手机号的。

```
messageID=$messageID&receiver=$number&state=$state&err_code=$code&event=$event
```

示例：

```
messageID=23B522F13F59CA1B-1-15B887C54B3-20000000A&receiver=11234567890&state=2&err_code=InvalidTemplateCode.Malformed|The specified templateCode is wrongly formed.&event=SendFailed
```

(2.2) 如果是账号类型错误只会返回一条异步信息，不包含手机号。

```
messageID=$messageID&state=$state&err_code=$account_err_code&event=$event
```

示例：

```
messageID=23B500F19959CA1B-1-15B212E97C0-20000000B&state=2&err_code=InvalidUserStatus.Malformed|The specified user status is wrongly formed.&event=SendFailed
```

(3) 回复消息

包括如下字段：

字段名	描述
sender	回复消息的人
content	回复消息的内容
receive_time	回复消息的时间
extend_code	用户定义的扩展CODE
ver	版本号
event	触发的事件类型

```
sender=$sender&content=$content&receive_time=$time&extend_code=$code&ver=$ver&event=$event
```

示例：

```
sender=8611234567890&content=刚才没开回信  
&receive_time=20170331132401&extend_code=111&ver=1.0&event=ReplyMessage
```

(三) 注意事项：

- 事件通知规则创建完，约10分钟左右生效
- 事件通知规则创建时，选择“主题”，与选择其他的接收终端是互斥的。即，如果选择了“主题”，那只能继续选择“主题”；如果选择了“非主题”的终端，那只能选择非主题的终端。
- 事件通知规则创建时，选择“主题”作为接收终端时，只能选择一个。
- 事件通知规则创建时，选择“主题”作为终端时，不能选择短信专用的主题（sms.xxx）。
- 事件通知规则创建时，如果选择“主题”作为接收终端时，用户需要自己去创建对应的订阅，定义自己的推送（消费）方式。
- 事件通知规则创建时，如果选择“非主题”作为接收终端时，会自动创建主题和对应的订阅。
- 事件通知规则支持创建和删除，暂时不支持修改

(四) 操作方式

(1) 创建事件通知规则

(1.1) 进入短信概览页



(1.2) 选择主题作为接收终端

创建规则

温馨提示： 您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示： 任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 规则名称 ? : 1. 设置规则名称

* 事件类型 ? : 2. 选择事件类型

温馨提示： 选择接收终端时，“主题”与其它选项互斥

* 接收终端 :

3. 选择接收终端

一旦选择了主题，
则不能继续选择其他终端
只能有一个主题接收终端

添加 您还可以添加 4 个接收终端

确认

取消

(1.3) 选择“非主题”作为接收终端

创建规则

温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 规则名称 ? :

* 事件类型 ? :

温馨提示：选择接收终端时，“主题”与其它选项互斥

如果选择了“非主题”类型的接收终端，就只能继续选择“非主题”类型的

* 接收终端 :

- http
- 队列
- 队列
- http

添加 您还可以添加 1 个接收终端

确认 取消

(2) 查看事件通知规则

(2.1) 事件通知规则列表

The screenshot shows the 'Message Service' console interface. On the right side, there is a table of event notification rules. The table has columns for '规则名称' (Rule Name), '主题' (Topic), and '事件通知' (Event Notification). The rule 'fail-rule-sample' is selected, and its details are shown in a modal window. The '事件通知' tab is active, showing a list of event types: 'fail-rule-sample' and 'success-rule-sample'. A red arrow points to the '事件通知' tab in the modal window.

(2.2) 事件通知规则细节



(3) 接收异步通知消息（队列）

当通过SDK发布一条消息后，可以根据对应事件规则中的描述，到指定的消息接收终端查看消息。比如创建规则时，指定将消息发送到队列中，那就可以在控制台中，直接到对应的队列中查询消息内容；如果是指定发送到用户的HTTP服务器上，用户的HTTP服务器会收到具体的推送消息（可参考：说明，SDK示例）。

(3.1) 如果创建规则时，选择的是主题，需要自行去创建订阅以及队列（已经有对应的订阅和队列的话，则跳过）

(3.1.1) 创建队列

(3.1.1.1) 进入队列标签页



(3.1.1.2) 创建队列

新建队列 ✕

1. 设置队列名称

* 队列名称 ? :

* 当前地域 : 华北2

消息接收长轮询等待时间(秒) ? :

取出消息隐藏时长(秒) ? :

消息最大长度(Byte) ? :

消息存活时间(秒) ? :

消息延时(秒) ? :

开启logging :

2. 点击确认

(3.1.2) 创建主题

(3.1.2.1) 进入主题标签页



(3.1.2.2) 创建主题



(3.1.3) 创建订阅

(3.1.3.1) 进入主题标签页



(3.1.3.2) 进入订阅标签页



(3.1.3.3) 创建订阅

新建队列



1. 设置队列名称

* 队列名称 ? :

queue-test-1

* 当前地域 :

华北 2

消息接收长轮询等待时间(秒) ? :

取出消息隐藏时长(秒) ? :

消息最大长度(Byte) ? :

消息存活时间(秒) ? :

消息延时(秒) ? :

开启logging :



2. 点击确认

确认

取消

(3.3) 从队列中接收消息

(3.3.1) 进入队列标签页

队列名称	消息生命周期(秒)	消息延时(秒)	消息消费数	未读消息数	延迟消息数	创建/最后修改时间	开启logging	操作
queue-test-1	360000	0	1	0	0	2017-03-31 16:31:26 2017-03-31 16:31:26	false	修改设置 删除消息 接收消息

(3.3.2) 接收消息

接收消息

✕

目的队列名称 :

消息句柄 [?] :

消息创建时间 [?] : 2017-03-31 17:06:00

被消费次数 [?] : 1

首次消费时间 [?] : 2017-03-31 17:07:49

下次消费时间 [?] : 2017-03-31 17:08:19

消息内容 [?] :

```
messageID=E0E958442232C064-1-15B2
39D3FC0-20000001&receiver=1826806
8778&state=2&err_code=InvalidDayuPar
amString.Malformed|The specified param
String is too long .&event=SendFailed
```

自动删除已接收消息 base64解码

警告!

- 1.如果当前操作的队列正被您的应用使用，请谨慎操作，以防您的应用消费异常。
- 2.如果复制未成功，可能是由于您未安装Flash或版本过低，请选择手动复制。

1. 取消base64编码

2. 继续接收消息

补充说明：使用SDK从对应Queue中接收异步回调结果消息的示例代码可参考：

- JavaSdk : [here](#)
- C++Sdk : [here](#)
- PythonSdk : [here](#)
- PhpSdk : [here](#)
- C#Sdk : [here](#)
- RestfulAPI : [here](#)

邮件推送事件通知

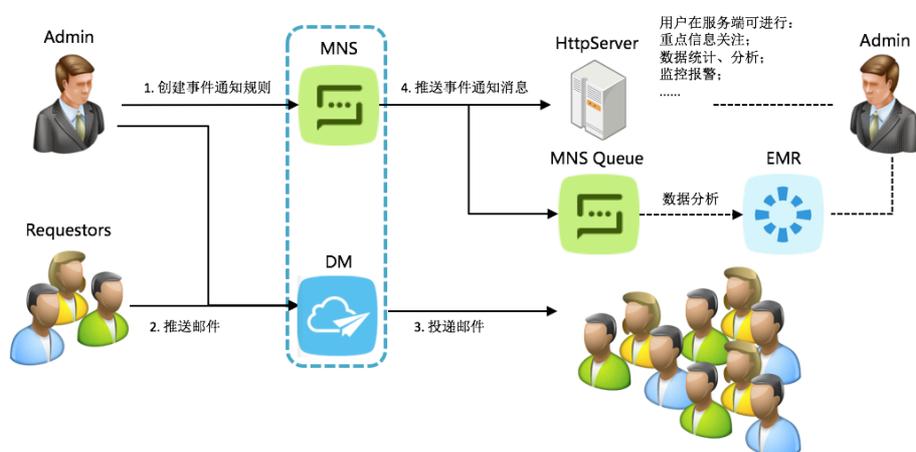
邮件推送服务提供了事件通知功能（可在其控制台上打开异步通知功能），支持将邮件推送的结果以消息的方

式异步通知给用户。用户可以通过创建事件通知的规则，来指定在邮件推送过程中触发不同的事件时，将邮件推送的结果消息推送到对应的接收端中。

可以解决的问题：

- 异步获取邮件推送的结果，再根据业务需求进行相应的处理，比如重新发送等等
- 对于大批量的邮件通知的场景，可异步统计推送情况
- 可以自动处理结果消息，对推送情况进行监控，比如发生突发的大量邮件推送失败，可以第一时间知晓
- 结果消息可以通过多种方式推送到不同的接收端（可能有多个业务部门均关心该数据），以进行不同的统计处理
- 如果A将推送的权限授予B，A可以通过该方式来知晓B推送短信的情况

示意图：



（一）事件通知规则组成

事件通知规则由两部分组成：

- 事件类型
- 消息接收终端

（1）事件类型

事件通知是按区域提供服务的，在创建规则时，关注某个（些）事件就是指关心这个区域的邮件推送过程中产生的所有该类型事件。邮件推送的事件有如下两种：

事件类型	描述
Deliver:SendSuccessfully	邮件推送成功
Deliver:SendFailed	邮件推送失败

（2）消息接收终端创建规则时，消息的接收终端可以有两种设置方式：

- 指定接收消息的Http服务或者MNS的队列：此时系统会自动帮忙创建一个主题，并且以指定的Http服务或者MNS队列作为Endpoint，创建相应的订阅。（如果事先没有主题，建议使用这种方式）
- 直接指定主题：此时，规则匹配的话，消息就会发布到主题中（此时用户可以选择已有的主题，或者先创建主题，再选择某个主题。**如果用户指定了不存在的主题，消息是没办法正常推送的**）。然后用户可以在该主题上创建订阅，指定通过何种方式将消息推送到指定的接收端。注：
 - **不要选择短信专用的主题（ sms.xxx ）作为消息接收端，短信专用Topic仅支持短信推送。**
 - **如果直接指定主题，只能指定一个。**

注：

事件通知规则指定消息接收端时，需要注意的是，**选择“主题”，与选择其他的接收终端是互斥的**。即，如果选择了“主题”，那只能继续选择“主题”；如果选择了“非主题”的终端，那只能选择“非主题”的终端。

补充：

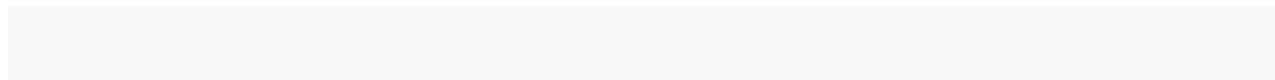
- 消息通知是基于MNS的主题实现的，支持多种推送方式，具体可参考：[here](#)。
- 接收消息的Http服务搭建可参考：[说明](#)，[SDK示例](#)。
- 从对应的队列中获取通知消息的方式可参考：[here](#)。

（二）邮件推送异步通知消息

邮件推送的结果消息格式如下（包含如下字段）：

字段名	描述
env_id	系统内ID（用于反馈查询问题）
msg_id	邮件原文中的Message-ID字段
account	发件人地址
from	发件人地址
rcpt	收件人地址
recv_time	邮件接收入队时间
end_time	邮件投递完成时间
status	投递完成结果状态（0-成功，2-无效地址，3-接收方认为垃圾邮件，4-失败）
event	触发的事件
region	时间规则所在区域
err_code	投递完成时接收方ESP返回码
err_msg	投递完成时接收方ESP返回信息

发送成功和发送失败时，消息中的字段都是一样的，格式如下：



```
env_id=$env_id&msg_id=$msg_id&account=$account&from=$from&rcpt=$rcpt&rcv_time=$time&end_time=$time&status=$status&event=$event&region=$region&err_code=$err_code&err_msg=$err_msg
```

错误说明：

错误信息请参考：[here](#)（重点关注其中 2.2 接收方常见退信代码）

此外，API级别的错误（MNS邮件订阅推送是出错）可参考：

- 单独推送：[here](#)
- 批量推送：[here](#)

消息示例

(1) 发送成功：

```
env_id=457323378&msg_id=77041400535408&account=test@test.com&from=test@test.com&rcpt=abc@abc.com&rcv_time=2017-04-14 11:44:47&end_time=2017-04-14 11:44:48&status=0&event=deliver&region=cn-hangzhou&err_code=250&err_msg=250 Send Mail OK
```

(2) 发送失败

```
env_id=457323379&msg_id=77041400535409&account=test@test.com&from=test@test.com&rcpt=abc@abc.com&rcv_time=2017-04-14 11:44:48&end_time=2017-04-14 11:44:49&status=2&event=deliver&region=cn-hangzhou&err_code=550&err_msg=550 No Such User
```

(三) 注意事项：

- 事件通知规则创建完，约10分钟左右生效
- 事件通知规则创建时，选择“主题”，与选择其他的接收终端是互斥的。即，如果选择了“主题”，那只能继续选择“主题”；如果选择了“非主题”的终端，那只能选择非主题的终端。
- 事件通知规则创建时，选择“主题”作为终端时，只能选择一个。
- 事件通知规则创建时，选择“主题”作为终端时，不能选择短信专用的主题（sms.xxx）。
- 事件通知规则创建时，如果选择“主题”作为接收终端时，用户需要自己去创建对应的订阅，定义自己的推送（消费）方式。
- 事件通知规则创建时，如果选择“非主题”作为接收终端时，会自动创建主题和对应的订阅。
- 事件通知规则支持创建和删除，暂时不支持修改

(四) 操作方式

(1) 创建事件通知规则

(1.1) 进入事件通知页



(1.2) 选择主题作为接收终端

创建规则

温馨提示： 您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示： 任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区： 华东 1

* 所在产品： 邮件推送 DM

* 规则名称： SendSuccessfully 1. 指定规则名称

* 事件类型： Deliver:SendSuccessfully x 2. 选择事件类型

温馨提示： 选择接收终端时，“主题”与其它选项互斥

* 接收终端： 主题 dm-result-topic-1 x

3. 选择接收主题

添加 您还可以添加 4 个接收终端

选择主题作为接收端时，
就只能选择一个主题

确认

取消

(1.3) 选择“非主题”作为接收终端

创建规则



温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

地区：华东 1

所在产品：邮件推送 DM

规则名称：SendFailed 1. 指定规则名

事件类型：Deliver:SendFailed x 2. 选择一个或多个事件类型

温馨提示：选择接收终端时，“主题”与其它选项互斥

接收终端：

http http://abc.com x

队列 dm-result-queue-1 x

队列 dm-result-queue-2 x

http http://def.com x

添加 您还可以添加 1 个接收终端

3. 选择接收终端

一旦选择“非主题”类型，就只能继续选择“非主题”类型

确认 取消

(2) 查看事件通知规则

(2.1) 事件通知规则列表

Message Service | 事件通知列表 华北 2 华东 1 华北 1 华东 2 华南 1 亚太东南 1 (新加坡) 刷新 创建规则

温馨提示：主题模型于2016-09-26开始正式收费，创建一条事件通知规则默认创建一个主题，可能会产生费用，详见产品价格

云产品通过MNS实现事件通知 (异步回调)、更多帮助。

产品名称：邮件推送 DM

规则名称	事件类型	接收终端	主题名称	操作
SendFailed	Deliver:SendFailed	dm-result-queue-1 dm-result-queue-2 http://abc.com http://def.com	mns-en-topics-dm-SendFailed-14...	删除
SendSuccessfully	Deliver:SendSuccessfully	acs.mns.cn-hangzhou:1996365277... acs.mns.cn-hangzhou:1996365277...	dm-result-topic-1,dm-result-to...	删除

1. 邮件推送 DM 对应的事件通知规则的列表

(3) 接收异步通知消息 (队列)

推送邮件后，可以根据对应事件规则中的描述，在指定的消息接收终端查看到结果消息。比如创建规则时，指定将消息发送到队列中，那就可以在控制台中，直接到对应的队列中查询消息内容；如果是指定发送到用户的 HTTP 服务器上，用户的 HTTP 服务器会收到具体的推送消息（可参考：说明，SDK 示例）。

(3.1) 如果创建规则时，选择的是主题，需要自行去创建订阅以及队列（已经有对应的订阅和队列的话，则跳过这两步）

(3.1.1) 创建队列

(3.1.1.1) 进入队列标签页



(3.1.1.2) 创建队列



(3.1.2) 创建主题

(3.1.2.1) 进入主题标签页



(3.1.2.2) 创建主题



(3.1.3) 创建订阅

(3.1.3.1) 进入主题标签页



(3.1.3.2) 进入订阅标签页



(3.1.3.3) 创建订阅

创建订阅



主题名称：

推送类型： 1. 选择推送类型
只允许推送到自己的队列

* 订阅名称： 2. 指定订阅名称

* 接收端地址： 3. 指定接收的队列名称

消息过滤标签：
用于消息过滤,不超过16个字符,当前可设置一个标签

* 重试策略： 退避重试 指数衰减重试

* 消息推送格式： SIMPLIFIED JSON XML

4. 点击确认

(3.2) 如果创建规则时，选择的是队列，则只需要自行创建队列即可（已有对应的队列，则跳过该步骤）

(3.2.1) 进入队列标签页

Message Service | 队列列表 | 华北2 | **华东1** | 香港 | 华北1 | 华东2 | 华南1 | 亚太东北1 (东京) | 亚太东南1 (新加坡) | 亚太东南2 (悉尼)

刷新 获取Endpoint **创建队列**

队列 **1. 选择队列标签** | 2. 选择区域 | 3. 点击创建队列

队列查询： 搜索

队列名称	消息生命周期(秒)	消息延时(秒)	活跃消息数	非活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作

(3.2.2) 创建队列

新建队列



* 队列名称 ? : 1. 指定队列名称

* 当前地域 : 华东 1

消息接收长轮询等待时间(秒) ? :

取出消息隐藏时长(秒) ? :

消息最大长度(Byte) ? :

消息存活时间(秒) ? :

消息延时(秒) ? :

开启logging :

确认

取消

(3.3) 从队列中接收消息

(3.3.1) 进入队列标签页

Message Service | 队列列表 | 华北 2 | **华东 1** | 香港 | 华北 1 | 华东 2 | 华南 1 | 亚太东北 1 (东京) | 亚太东南 1 (新加坡) | 亚太东南 2 (悉尼) | 刷新 | 获取Endpoint | 创建队列

队列 | 欧洲中部 1 (法兰克福) | 中东东部 1 (迪拜) | 英国西部 1 (硅谷)

队列名称: 搜索

队列名称	消息生命周期(秒)	消息延时(秒)	活跃消息数	非活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作
dm-result-queue-1	345000	0	2	0	0	2017-04-14 09:37:26	false	修改设置 删除 接收消息
						2017-04-14 09:37:26		发送消息

1. 点击接收消息

(3.3.2) 接收消息

接收消息

✕

目的队列名称 :

dm-result-queue-1

消息句柄 ? :

1-ODU4OTkzNDU5My0xNDkyMTC

复制

消息创建时间 ? :

2017-04-14 11:08:03

被消费次数 ? :

1

首次消费时间 ? :

2017-04-14 11:22:05

下次消费时间 ? :

2017-04-14 11:22:35

消息内容 ? :

```
env_id=12654662927&msg_id=c937403c-56bc-406c-a1de-620371e820d3@alibaba.com&account=mnstest@mnstest.css9.net&from=mnstest@mnstest.css9.net&rcpt=nightleslie@gmail.com&recv_time=20
```

 自动删除已接收消息 base64解码

警告!

1. 取消base64解码

- 1.如果当前操作的队列正被您的应用使用，请谨慎操作，以防您的应用消费异常。
- 2.如果复制未成功，可能是由于您未安装Flash或版本过低，请选择手动复制。

2. 继续接收消息

接收消息

取消

补充说明：使用SDK从对应Queue中接收异步回调结果消息的示例代码可参考：

- JavaSdk : [here](#)
- C++Sdk : [here](#)
- PythonSdk : [here](#)
- PhpSdk : [here](#)
- C#Sdk : [here](#)
- RestfulAPI : [here](#)

日志管理

MNS 的日志管理功能将用户的消息操作日志推送到指定 LoggingBucket 中。

用户在控制台上配置将日志推送到 OSS 或者 LogService，然后开启该地域队列/主题的日志管理功能，MNS 将自动推送该队列/主题消息的操作日志到指定的 LoggingBucket 中。

推送日志到LogService(必读)

- 配置方法：推送日志到LogService
- 日志查看：查看日志——LogService
- 在为LogStore创建索引时，请选择合理的数据保存时间，该属性后续只支持缩短，不支持延长；
- 如果用户将 LoggingBucket 对应的LogService 的Project、LogStore删除，或者将授予 MNS 的权限取消，日志将无法正常推送给用户；
- 日志延迟时间约5分钟；

推送日志到OSS(必读)

- 配置方法：推送日志到OSS
- 日志查看：查看日志——OSS
- 如果用户将 LoggingBucket 对应的OSS Bucket删除，或者将授予 MNS 的权限取消，日志将无法正常推送给用户；
- 日志延迟时间约15分钟；

细节分析

- 每个地域配置一个 LoggingBucket，该地域所有开通日志管理功能的队列/主题的消息操作日志均推送到该 LoggingBucket中；
- 每个队列/主题可以独立设置是否开启日志管理功能，默认不开启；

费用解析

- MNS 不针对日志管理功能收取额外费用；
- MNS 将日志推送到OSS 或者 LogService中，对应服务会根据存储空间、流量、请求数等进行收费，具体细则请参考：LogService价格总览、OSS价格总览
- 日志量的大小与用户操作 MNS 的 Qps 和操作类型有关系，此处以 SendMessage 操作、Qps 1000为例，计算每分钟的日志量： $178\text{Byte} \times 1000 \times 60 / 1024 / 1024$ 约为 10MB，其中 178 为单条 SendMessag操作的日志大小。

队列消息操作日志

队列消息操作日志是指操作队列消息所产生的日志，比如发送消息、消费消息、删除消息等操作。

一条消息操作日志中包含多个字段，每个字段都有自己的含义。

根据操作的不同，消息操作日志所包含的字段也不相同。

接下来，将分别介绍各个字段的含义和不同操作所包含的字段信息。

日志字段解析

一条消息操作日志中包含多个字段，各个字段的含义如表格所示。

字段	含义
Time	本次操作的发生时间
MessageId	消息的 MessageId，标识本次操作处理的消息
QueueName	本次操作对应的队列名称
AccountId	本次操作对应队列的账号
RemoteAddress	发起该操作的客户端地址
NextVisibleTime	该操作执行完成后，这条消息的下次可见时间
ReceiptHandleInRequest	用户执行该操作时传入的 ReceiptHandle 参数
ReceiptHandleInResponse	该操作执行完成后，返回给用户的 ReceiptHandle

各个操作的字段列表

不同操作的日志包含的字段信息各不相同，具体每个操作包含的字段请参考表格。

操作	Time	Queue Name	AccountId	MessageId	RemoteAddress	NextVisibleTime	ReceiptHandleInResponse	ReceiptHandleInRequest
SendMessage/BatchSendMessage	有	有	有	有	有	有	-	-
PeekMessage/BatchPeekMessage	有	有	有	有	有	-	-	-
ReceiveMessage/BatchReceive	有	有	有	有	有	有	有	-

Message								
ChangeMessageVisibility	有	有	有	有	有	有	有	有
DeleteMessage/BatchDeleteMessage	有	有	有	有	有	有	-	有

主题消息操作日志

主题消息操作日志是指操作主题消息产生的日志，主要有两类：发布消息和推送消息。接下来，将介绍主题消息操作日志各个字段的含义，以及不同的操作所包含的字段信息。

日志字段解析

一条消息操作日志中包含多个字段，各个字段的含义如表格所示。

字段	含义
Time	本次操作的发生时间
MessageId	消息的 MessageId，标识本次操作处理的消息
TopicName	本次操作对应的主题名称
SubscriptionName	本次操作对应的订阅名称
AccountId	本次操作对应主题的账号
RemoteAddress	发起该操作的客户端地址
NotifyStatus	MNS 将消息推送给用户时，用户返回的状态码或者相应的出错信息

各个操作的字段列表

不同操作的日志包含的字段信息各不相同，具体每个操作包含的字段请参考表格。

操作	Time	MessageId	Topic Name	Subscription Name	AccountId	RemoteAddress	Notify Status	Subscription Name
PublishMessage	有	有	有	-	有	有	-	-

Notify	有	有	有	有	有	-	有	有
--------	---	---	---	---	---	---	---	---

NotifyStatus

NotifyStatus 是推送消息日志特有的字段，该可以协助您调查MNS 推送消息到 Endpoint失败的原因。根据不同的 NotifyStatus，您可以按照下表建议的处理方法进行处理。

错误码	描述	建议处理方法
2xx	消息推送成功	无
其它Http状态码	消息推送给用户，Endpoint 返回了非2xx的状态码	检查 Endpoint 端处理逻辑
InvalidHost	订阅指定的 Endpoint 不合法	确认订阅中 Endpiont 是否真实有效，可使用curl/telnet进行确认
ConnectTimeout	连接订阅指定的 Endpoint 超时	确认订阅中 Endpoint 当前是否可访问，可使用curl/telnet进行确认
ConnectFailure	连接订阅指定的 Endpoint 失败	确认订阅中 Endpoint 当前是否可访问，可使用curl/telnet进行确认
UnknownError	未知错误	请联系 MNS 技术人员支持

日志管理功能可以做什么？

- 消息成功发送到队列，但消费端收不到消息，消息去哪儿了？
- 消息被谁消费了、消费了几次？
- 消费端宕机了，消费失败的消息什么时候可以再次被消费？
- 消息发布到主题，Endpoint 却迟迟收不到消息？
- 想看一个月前的消息操作日志？

MNS 的日志管理功能通通满足你！！

- 将日志推送到LogService，控制台查看完整消息轨迹：[查看日志——LogService](#)
- 使用官网提供查询工具，指定相应参数即可查看消息的处理日志；[日志查询命令行工具](#)
- 登陆OSS控制台，配置LoggingBucket的LifeCycle属性，查看一年前的日志也不是问题：[oss控制台](#)
- 除了官方工具，还可以写写代码，通过 OSS 的 GetObject 接口下载日志文件，做你想做的任何事情；

邮件推送

MNS消息服务的Topic目前已经支持邮件推送，用户发送到Topic里面的Message不仅可以被推送到HttpServer和MNS的Queue，现在也可以被同时推送到邮箱。

Endpoint格式

mail:directmail:XXX@YYY.com，其中"XXX@YYY.com"是您的邮箱地址。

前置条件

只需要先开通阿里云的邮件推送服务即可，具体开通方式请参考下面文档。

开通邮件推送服务

1. 确认已拥有在域名服务商注册过的域名，一般可以使用您公司的域名 点此查询和购买域名 以万网为

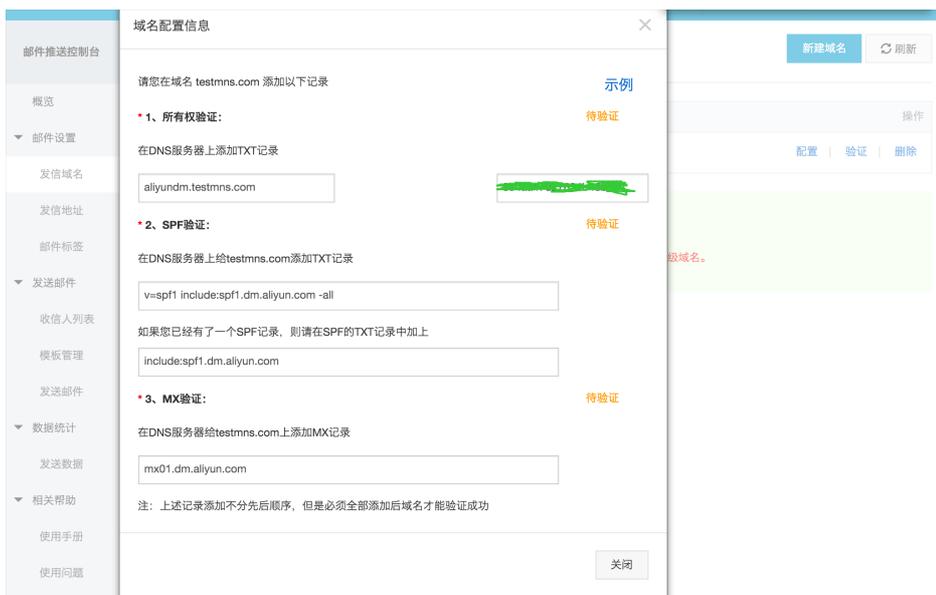


例：

2. 进入邮件推送控制台
3. 在邮件推送控制台配置域名



点击配置后会看到以下信息，会在第4步用到这些信息：



4. 将配置信息填写到第一步时确认的域名服务商处，具体填写方法请参考域名解析配置 快捷链接：万网域名控制台
5. 在邮件推送控制台配置发信地址



6. 现在，您已经可以完成配置，可以在邮件推送控制台看到发信地址了。



控制台体验流程

创建邮件订阅

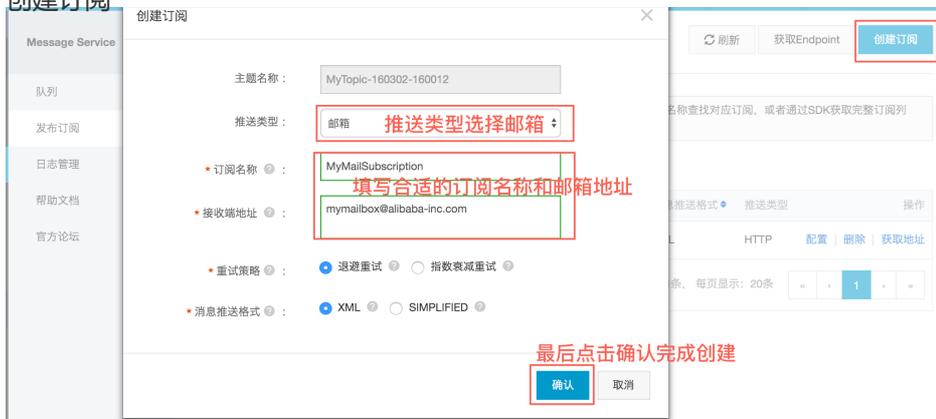
1. 在MNS控制台创建Topic



2. 进入Topic的"订阅详情"



3. 创建订阅



发送消息 (PublishMessage)

1. 进入Topic列表页面

2. 向Topic发送消息



3. 配置邮件消息的相关属性



邮件推送的相关属性可以参考这里



4. 发送消息，并查收邮件

JavaSDK订阅代码示例

```
CloudAccount cloudAccount = new CloudAccount(ACCESS_ID, ACCESS_KEY, ENDPOINT);
MNSClient client = cloudAccount.getMNSClient();
```

```
CloudTopic topic = null;
try {
    // 1. first create the topic
    String topicName = "TOPICNAME";
    TopicMeta meta = new TopicMeta();
    meta.setTopicName(topicName);
    topic = client.createTopic(meta);
}
```

```
// 2. generate the mail endpoint
String mailEndpoint = topic.generateMailEndpoint("mymailbox@alibaba-inc.com");

// 3. now subscribe to topic
SubscriptionMeta subMeta = new SubscriptionMeta();
subMeta.setSubscriptionName(SUBNAME);
subMeta.setEndpoint(mailEndpoint);

topic.subscribe(subMeta);

// 4. publish message
RawTopicMessage msg = new RawTopicMessage();
msg.setMessageBody("hello topic");

// 4.1 set the necessary attributes for mail
MessageAttributes messageAttributes = new MessageAttributes();
MailAttributes mailAttributes = new MailAttributes();
mailAttributes.setAccountName("direct_mail_account_name@aliyun-inc.com");
mailAttributes.setSubject("TestMailSubject");
messageAttributes.setMailAttributes(mailAttributes);

// 4.2 publish
TopicMessage msg2 = topic.publishMessage(msg, messageAttributes);
System.out.println(msg2.getMessageId());
System.out.println(msg2.getMessageBodyMD5());
} catch (ServiceException se) {
    System.out.println(se.getErrorCode() + se.getRequestId());
    System.out.println(se.getMessage());
    se.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}
```

Http Request Body示例

```
POST /topics/$TopicName/messages HTTP/1.1
Host: $AccountId.mns.cn-hangzhou.aliyuncs.com
Date: Tue, 08 Dec 2015 06:13:40 GMT
Content-Length:500
Content-Type:text/xml;charset=utf-8
Authorization: MNS 15B4D3461F177624206A:aibzWu1iDEx9LwO56+kHgA3eqmI=
x-mns-version: 2015-06-06

<?xml version="1.0" encoding="UTF-8"?>
<Message xmlns="http://mns.aliyuncs.com/doc/v1">
  <MessageBody>hello topic</MessageBody>
  <MessageAttributes>
    <DirectMail>{"Subject":"TestMailSubject","AccountName":"direct_mail_account_name@aliyun-
inc.com","ReplyToAddress":0,"AddressType":0,"IsHtml":0}</DirectMail>
  </MessageAttributes>
</Message>
```

Queue推送

用户使用已创建Queue作为Endpoint对Topic内容进行订阅；当有消息发布到Topic中时，Topic会通过内部数据通道，将Topic中的消息推送到各个订阅的Queue中。用户只需对订阅的Queue进行轮询拉取即可。

使用场景

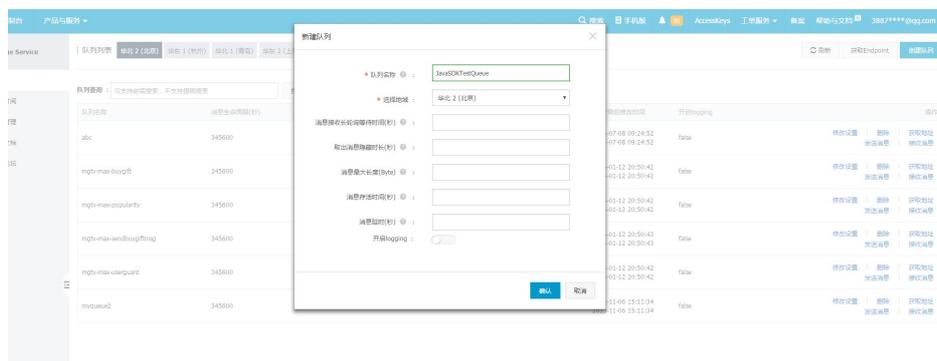
1. 订阅者无需保持在线状态；
2. 订阅者机器部署在VPC环境；
3. 订阅者需要通过高并发提高消费速度；

控制台订阅步骤

1. 创建主题Topic



2. 在同一地域创建队列Queue



3. 选中已创建的Topic，点击“订阅详情”进入订阅管理页面



4. 点击“创建订阅”，选择推送类型为“队列”，同时输入已创建的队列名（此处并不会对队列是否存在做检测，请确保相应的队列已经创建）



Java SDK订阅代码示例

```

client = new DefaultMNSClient(ServiceSettings.getMNSEndpoint(),
    ServiceSettings.getMNSAccessKeyId(),
    ServiceSettings.getMNSAccessKeySecret());

// 创建队列Queue
String queueName = "JavaSDKTestQueue";
String subName = "JavaTestSub"
QueueMeta queueMeta = new QueueMeta();
queueMeta.setQueueName(queueName);
CloudQueue queue = client.createQueue(queueMeta);
// 创建主题Topic
String topicName = "SampleTopic";
TopicMeta meta = new TopicMeta();
meta.setTopicName(topicName);
topic = client.createTopic(meta);
// 使用队列作为endpoint进行订阅
String queueEndpoint = topic.generateQueueEndpoint(queueName);
SubscriptionMeta subMeta = new SubscriptionMeta();
subMeta.setSubscriptionName(subName);
subMeta.setNotifyStrategy(SubscriptionMeta.NotifyStrategy.EXPONENTIAL_DECAY_RETRY);
subMeta.setEndpoint(queueEndpoint);
String subUrl = topic.subscribe(subMeta);

```

短信推送

短信发送完整流程

请点击[这里](#)跳转

移动推送

简介

MNS消息服务的Topic目前已经支持移动推送，用户发送到Topic里面的Message可以同时被阿里云的移动推送服务推送到移动端。

Endpoint格式

push:{APP_KEY}，其中{APP_KEY}是移动端的App在阿里云的移动推送服务的注册ID

前置条件

只需要先开通阿里云的移动推送服务即可，具体开通方式请参考下面文档。

配置移动推送服务

创建App：在移动推送控制台(<https://push.console.aliyun.com/>)的App列表页，点击页面右上角的“创建APP”按钮即可创建一个新的App。



创建APP

* APP名称: 阿里云推送App

分类: 其他

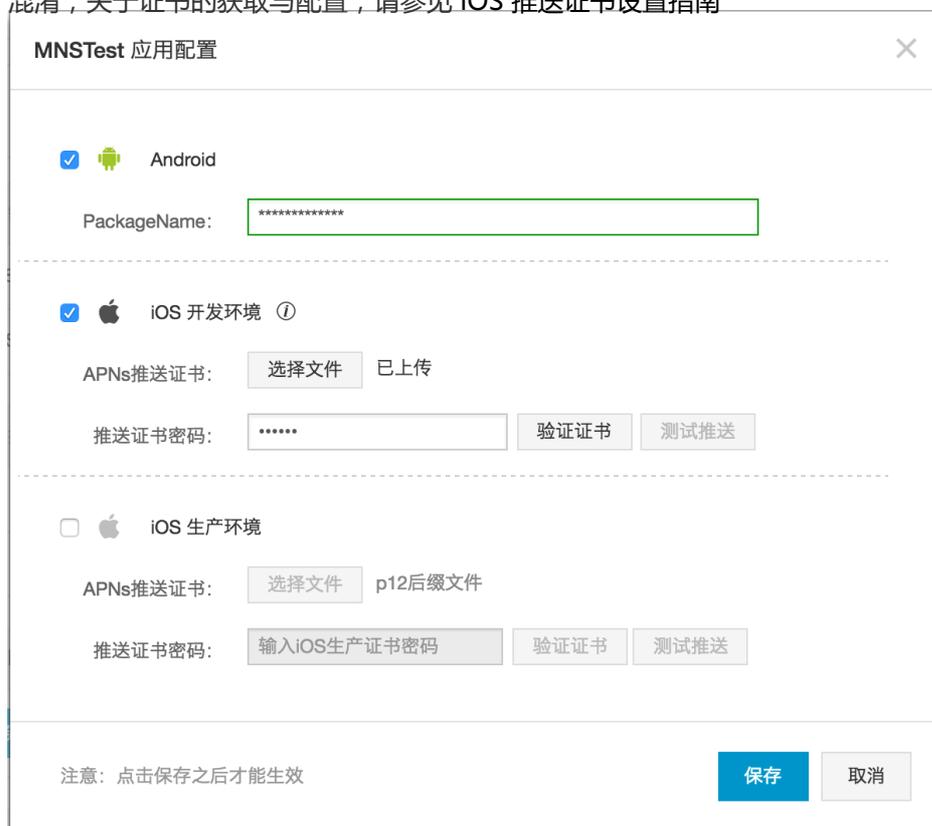
描述: 示例App

我已阅读并同意《移动服务用户服务条款》

创建 取消

配置App：App创建成功后，强烈建议您去配置app。在APP详情页点击配置应用。

- Android配置：勾选Android，并填入PackageName，包名需要与您的代码中的applicationID保持一致。请务必确保您创建的不同应用具有不同的PackageName，阿里云移动推送Android SDK以PackageName作为路由依据，不同应用配置相同包名，可能导致消息错误发送，即与目标应用有相同包名的应用接收到推送消息。
- iOS配置：勾选iOS，上传证书并填写证书对应的密码，注意 开发证书和生产证书 不要弄混淆，关于证书的获取与配置，请参见 [iOS 推送证书设置指南](#)



MNSTest 应用配置

Android

PackageName: *****

iOS 开发环境 ⓘ

APNs推送证书: 选择文件 已上传

推送证书密码: 验证证书 测试推送

iOS 生产环境

APNs推送证书: 选择文件 p12后缀文件

推送证书密码: 输入iOS生产证书密码 验证证书 测试推送

注意：点击保存之后才能生效

保存 取消

在APP列表点击应用证书按钮即可查看该应用的Appkey和Secret。



在APP列表页下载App端需要集成的SDK



配置SDK

- iOS 版 SDK 请参考文档：iOS SDK QuickStart
- Android 版 SDK 请参考文档：Android SDK QuickStart

安装配置好的App到终端手机

MNS控制台体验流程

创建移动推送订阅

1. 在MNS控制台创建Topic



2. 进入Topic的“订阅详情”





3. 创建订阅

发送消息 (PublishMessage)

1. 进入Topic列表页面



2. 向Topic发送消息

3. 配置移动推送消息的相关属性

发布消息 [← 返回主题列表](#)

注意: 如果当前操作的主题正被您的应用使用, 请谨慎操作, 以防您的应用处理异常。

主题名称:

* 消息内容 :

推送到其他终端时可能用到的MessageBody可以在这里填写

推送类型: 队列 http 阿里短信 邮箱 移动推送

消息过滤标签:

需要同时推送到移动端, 则需要选中这里

用于消息过滤, 不超过16个字符, 当前可设置一个标签

移动推送的具体属

移动推送设置 ✕

* 推送目标类型:

全部

 此处多种类型可选

设备类型:

推送类型:

* 标题 :

* 内容:

定时发送 :

是否保存离线消息 :

iOS通知配置: [查看](#)

Android通知配置: [查看](#)

配置好之后, 点击确认, 即可在手机上查看到推送的消息

性可以参考 [这里](#)

JavaSDK订阅代码示例

```
try {
    String topicName = TOPICNAME;
    TopicMeta meta = new TopicMeta();
    meta.setTopicName(topicName);

    CloudTopic topic = client.createTopic(meta);
    String pushEndpoint = topic.generatePushEndpoint("XXXX");

    SubscriptionMeta subMeta = new SubscriptionMeta();
    subMeta.setSubscriptionName(SUBNAME);
    subMeta.setNotifyStrategy(SubscriptionMeta.NotifyStrategy.EXPONENTIAL_DECAY_RETRY);
    subMeta.setEndpoint(pushEndpoint);
    topic.subscribe(subMeta);

    RawTopicMessage msg = new RawTopicMessage();
    msg.setMessageBody("hello topic");

    MessageAttributes messageAttributes = new MessageAttributes();
    PushAttributes pushAttributes = new PushAttributes();
    pushAttributes.setTarget(PushAttributes.PushTarget.ALL);
    pushAttributes.setTargetValue("ALL");
    pushAttributes.setDeviceType(PushAttributes.PushDeviceType.ALL);
    pushAttributes.setTitle("JavaSDK");
    pushAttributes.setBody("Hello From JavaSDK");
    pushAttributes.setPushType(PushAttributes.PushType.MESSAGE);
    pushAttributes.setParam("AndroidOpenType", "APPLICATION");
    messageAttributes.setPushAttributes(pushAttributes);

    topic.publishMessage(msg, messageAttributes);
} catch (ServiceException se) {
    System.out.println(se.getErrorCode() + se.getRequestId());
    System.out.println(se.getMessage());
    se.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}
```

云监控【主题模型】

MNS主题模型已经接入云监控，用户可以通过云监控针对主题级别（在云监控中，一个主题为一个实例）对如下指标实现统计，以及监控，并且可以设置相应的报警。指标包括：

- 指定时间内的消息推送次数
- 指定时间内的消息推送成功次数

- 指定时间内的消息推送成功率

使用场景

通过对这些指标的监控，可以在下述（不限于）场景使用：

如果用户业务稳定，比如每天都会有固定的消息推送量，那可以通过监控（设置报警）消息推送的总次数（也可以关注推送成功的次数）。如果低于某个值，可能就说明用户的业务出现异常，这种情况下可以有报警来提醒用户关注下自己的业务。

用户提供的 Endpoint（比如 HttpServer）可能会出现异常，比如 HttpServer 可能会 Crash，网络可能会出现异常等等。这种情况下会出现推送错误，此时可以通过监控（设置报警）推送成功率，当低于某个阈值时，就能够提醒用户关注接收消息的 Endpoint 是否出现了异常。

使用方法

1. 找一个需要监控的主题为例

1.1 如下图所示：进入MNS控制台，点击左侧“发布订阅”的标签，再选择一个需要监控的主题（比如图中的“华北2”区域的 topic-show-cloud-monitor），可以点击订阅详情，查看其所有的订阅关系。



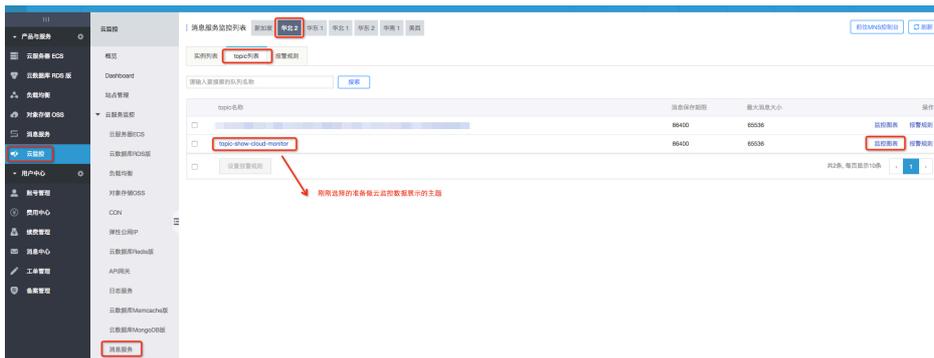
1.2 如下图所示：该主题下有两个订阅，分别为 valid-sub-show-cloud-monitor（有效的订阅，订阅时设置的接收消息的 Endpoint 是存在的 Queue，会100% 推送成功的）和 invalid-sub-show-cloud-monitor（无效的订阅，订阅时设置的接收消息的 Endpoint 是不存在的 Queue，所以都会推送失败），这两个订阅这样设置，是为了更好的展示云监控的数据。



2. 找到被监控主题对应的云监控实例

2.1 如下图所示，在控制台左侧的产品与服务中选择“云监控”，或者在最上方“产品与服务”中选择“云监控”，也可以直接点击云监控控制台进入。然后选择中间一栏的“消息服务”，进入消息服务MNS的标签页。再在右侧选择对应的区域（本示例为“华北2”），再选择“Topic列表”标签，最好在Topic列表中选择第一步

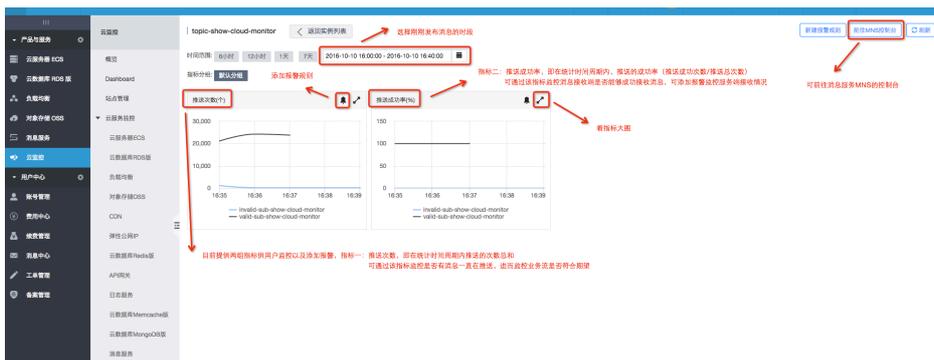
中想要监控的主题。



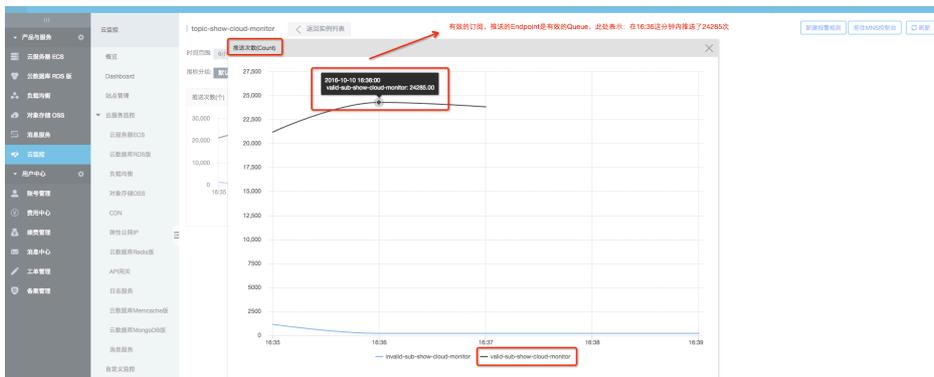
3. 查看监控数据

3.1 如下图所示，监控图表中有两张图：“推送次数”和“推送成功率”，分别对应消息服务的两个监控指标。可以选择时间范围，进行查看（图中选择了示例的16:00-16:40，该时段有消息推送）。小图中可以点击右上角的“警铃”设置报警规则，也可以点击最右方的图表查看大图。

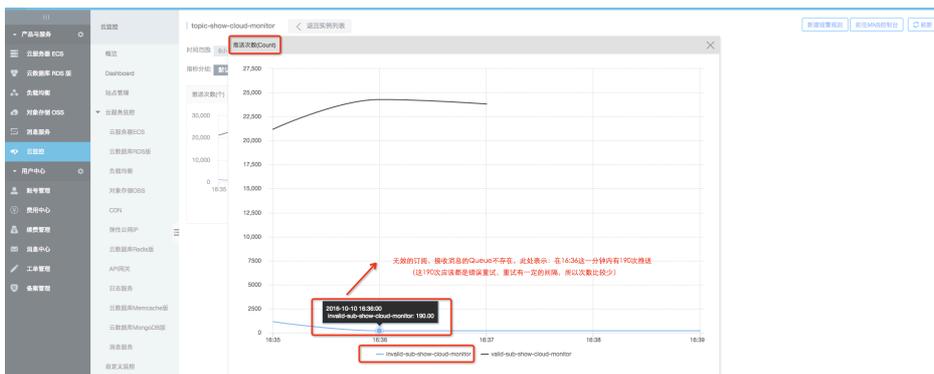
- 推送次数，即：在统计时间周期内，推送次数的总和，目前每分钟都有数据采集。
- 推送成功率，即：在统计时间周期内，推送的成功率（推送成功次数/推送总次数）



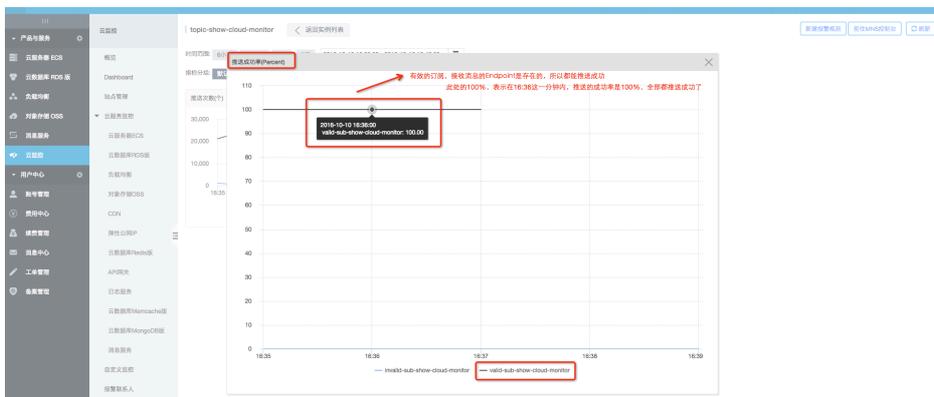
3.2 如下图所示，点击查看推送次数的大图，其中有两条线，分别对应不同的 Subscription，此处黑色的表示 valid-sub-show-cloud-monitor，从图中可以看到，在16:36分，一共推送了24285次。



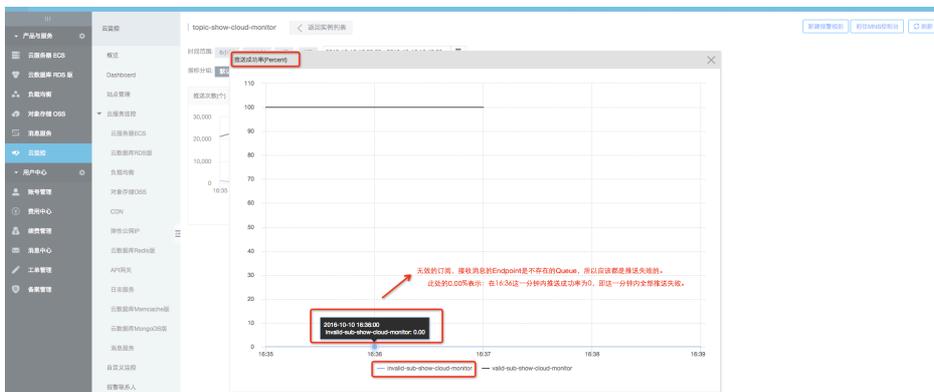
3.3 如下图所示，仍然是推送次数的大图，其中的蓝线表示 invalid-sub-show-cloud-monitor，从图中可以看到，在16:36分一共推送了190次。这190次应该都是在错误重试。重试会一定的时间间隔，所以次数比较少。



3.4 如下图所示，点击查看推送成功率的大图，图中也是两条线，也是分别对应两个不同的 Subscription。此处黑色的线表示 valid-sub-show-cloud-monitor，从图中可以看出，在16:36分，推送的成功率为100%，即所有的推送都是成功的。



3.5 如下图所示，仍然是推送成功率的大图，其中的蓝线表示 invalid-sub-show-cloud-monitor，从图中可以看到，在16:36分，推送的成功率为0%，即所有的推送都失败了。



4. 设置报警

4.1 如下图所示，点击右方“新建报警规则”，或者点击监控图表中的“警铃”图标，都可以弹出设置报警规则的弹窗。可以批量进行创建报警规则。可以点击“添加报警规则”来创建多条报警规则。图中设置了三条报警规则：

- 报警规则一表示：每5分钟读取一次1分钟内的推送次数，如果小于1000次的情况出现1次，就会触发报警。

- 报警规则二表示：每5分钟读取一次1分钟内的推送成功次数，如果小于800次的情况连续出现3次，就会触发报警。
- 报警规则三表示：没5分钟读取一次1分钟内的推送成功率，如果低于80%的情况连续出现2次，就会触发报警。



4.2 报警设置完成后，点击下一步设置通知对象（云监控中的联系人），可以通过短信通知用户等等。操作方法跟云监控其他设置报警的方法一致。

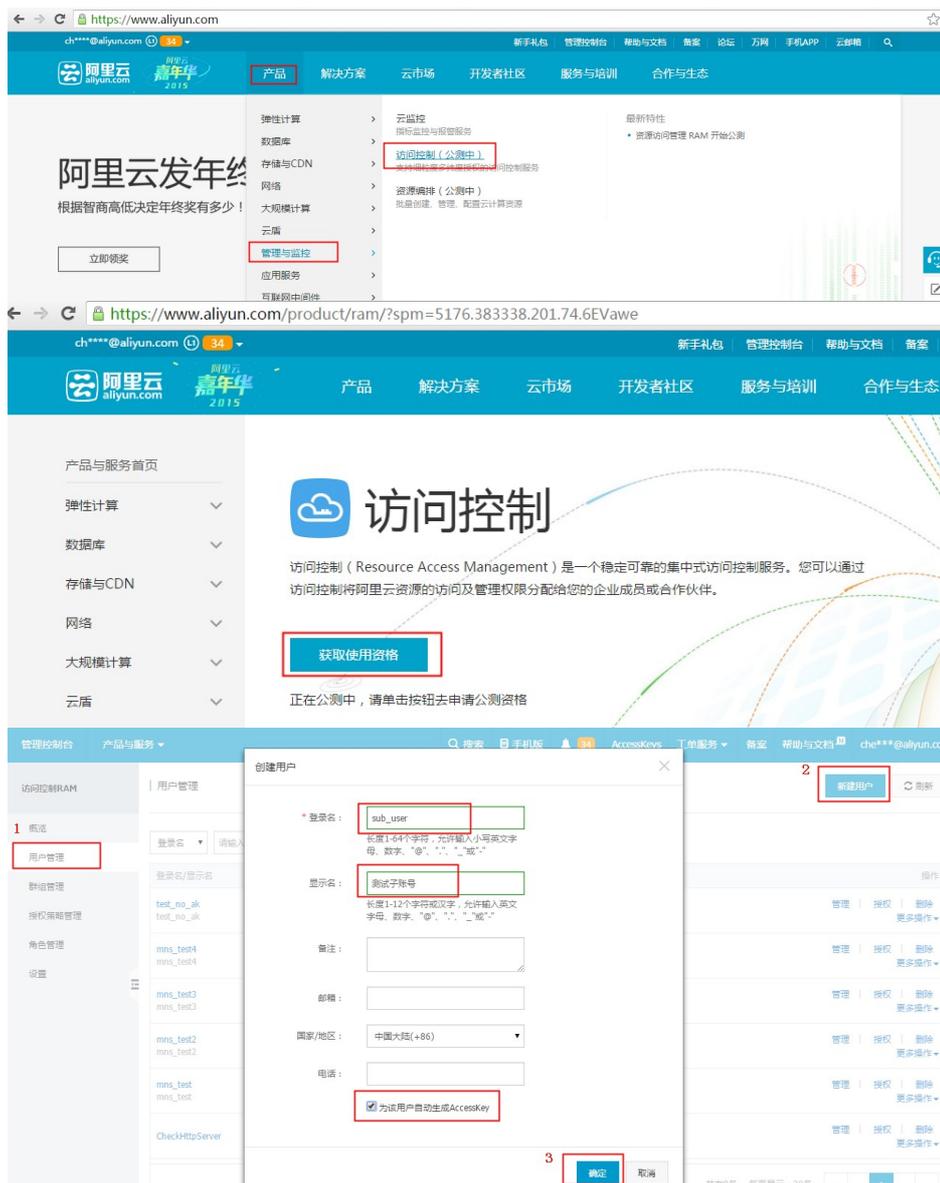
控制台操作指南

子账号访问MNS

[1]创建子账号

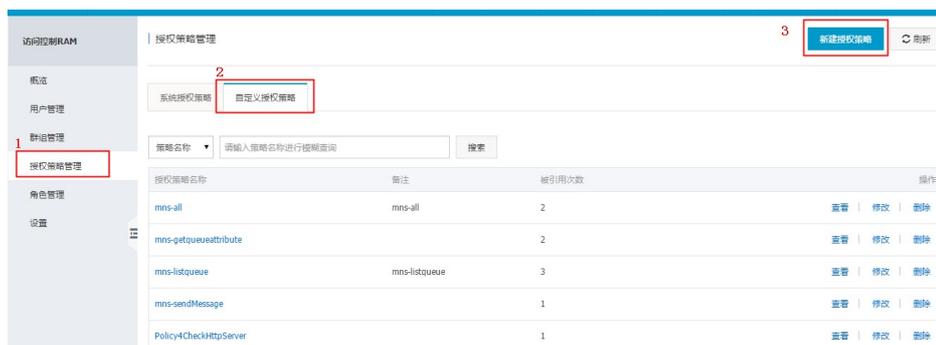
- 1 用主账号登陆阿里云官网
- 2 进入控制台：依次将鼠标移动至 **产品** -> **管理与监控**，单击 **访问控制** 进入访问控制产品页；单击 **获取使用资格** 申请使用权限，然后单击 **管理控制台** 进入控制台；
- 3 依次单击 **用户管理** -> **新建用户**（右上角）进入新建用户页面
- 4 在创建用户对话框里输入：登陆名（必填）和显示名，备注，邮箱，电话等（可选），同时勾选：**为该用户自动生成AccessKey**
- 5 点击 **确定**，完成子账号创建。创建成功后，会提示是否下载子账号 AK，请选择下载并妥善保管子账号 AK 文件。

图示1：



[2]新建授权策略

- 1 在 RAM 控制台上依次点击：**授权策略管理** -> **自定义授权策略**-> **新建授权策略**
 图示1



- 2 在弹出对话框中：选择授权策略模板（使用空模板）
- 3 编辑授权策略并提交：修改 **授权策略名称**（自定义名称），**备注**，**策略内容**，并提交。

示例：一个 MNS 授权策略内容模板：

```
{
  "Statement": [
    {
      "Action": "mns:*",
      "Effect": "Allow",
      "Resource": "acs:mns:*:*:*"
    }
  ],
  "Version": "1"
}
```

授权策略是json格式的字符串，其中，

Action：表示要授权的操作，MNS 操作都以"mns:"开头，

例如: "mns:SendMessage" 表示 MNS 服务的API：SendMessage/BatchSendMessage

其他详见附录：MNS API和授权操作映射表；

Effect：表示授权类型，例如:Allow, Deny

Resource：表示要授权的阿里云资源名（ARN），格式为："acs:<云服务名>:<地域名>:<主账号UID>:<资源URI>"

例如："acs:mns:cn-hangzhou:123456789:/queues/MyQueue1/messages"

表示：授权资源是主账号UID为123456789在cn-hangzhou地域的MyQueue1。

用"*"表示不指定具体的字段，例如："acs:mns:*:*:*" 表示不指定地域名,主账号ID和资源URI，子账号可以访问主账号的所有mns资源。

详细授权策略的规则说明请参考RAM帮助文档

图示2



[3]授权子账号访问MNS

- 1.1 返回 **用户管理**，找到第一步创建的子账号，点击右侧 **授权**
- 1.2 在弹出的对话框中，选择授权策略名称，并添加到右侧已选授权策略列表，点击 **确定** 提交

图示3：



2.使用子账号进行访问 MNS 服务

- 2.1 下载最新版本 mns python sdk 到 ECS 服务器并解压
- 2.2 进入 SDK 代码目录，修改sample.cfg，配置步骤1中下载的子账号 AK (AccessKeyId, AccessKeySecret) 和主账号的 Endpoint
- 2.3 运行sample 命令: \$python sample.py
如果没有报错，说明子账号 AK 能够成功访问主账号的 queue 。

[4]授权策略示例

1. 如何使用带IP限制的访问控制

示例1：在Allow授权中增加IP限制

允许通过42.120.88.0/24, 42.120.66.0/24两个IP段访问MNS

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": "mns:*",
      "Effect": "Allow",
      "Resource": "acs:mns:*:*",
      "Condition": {
        "IpAddress": {
          "acs:SourceIp": ["42.120.88.0/24", "42.120.66.0/24"]
        }
      }
    }
  ]
}
```

示例2：在Deny授权中增加IP限制

如果源IP不在42.120.88.0/24中则禁止对MNS执行任何操作；

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": "mns:*",
      "Effect": "Deny",
      "Resource": "acs:mns:*:*",
      "Condition": {
        "NotIpAddress": {
          "acs:SourceIp": ["42.120.88.0/24"]
        }
      }
    }
  ]
}
```

注意：因为Policy的鉴权规则是Deny优先(即如果用户的访问操作命中任意一条Deny规则，则禁止访问)，所以访问者从42.120.88.0/24以外的IP地址访问时，MNS服务会报没有权限。

示例3：授予MNS控制台可读权限

仅允许子账号拥有查看队列、主题属性以及列表的权限

```
{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "mns:ListQueue",
        "mns:ListTopic",
        "mns:GetQueueAttributes",
        "mns:GetTopicAttributes"
      ],
      "Resource": "acs:mns:*:*:*"
    }
  ]
}
```

[5]MNS API授权映射表

- 为保证安全，在添加子账号，添加授权策略和增加授权时，需要手机短信或邮件验证。

MNS API授权映射表

MNS API	RAM 授权操作	资源 (Resource)
ListQueue	mns:ListQueue	acs:mns:\$region:\$accountid:/queues
CreateQueue	mns>CreateQueue	acs:mns:\$region:\$accountid:/queues/\$queueName
DeleteQueue	mns>DeleteQueue	acs:mns:\$region:\$accountid:/queues/\$queueName
SetQueueAttributes	mns:SetQueueAttributes	acs:mns:\$region:\$accountid:/queues/\$queueName
GetQueueAttributes	mns:GetQueueAttributes	acs:mns:\$region:\$accountid:/queues/\$queueName

SendMessage/BatchSendMessage	mns:SendMessage	acs:mns:\$region:\$accountid:/queues/\$queueName/messages
ReceiveMessage/BatchReceiveMessage	mns:ReceiveMessage	acs:mns:\$region:\$accountid:/queues/\$queueName/messages
DeleteMessage	mns>DeleteMessage	acs:mns:\$region:\$accountid:/queues/\$queueName/messages
PeekMessage/BatchPeekMessage	mns:PeekMessage	acs:mns:\$region:\$accountid:/queues/\$queueName/messages
ChangeMessageVisibility	mns:ChangeMessageVisibility	acs:mns:\$region:\$accountid:/queues/\$queueName/messages
ListTopic	mns:ListTopic	acs:mns:\$region:\$accountid:/topics
CreateTopic	mns>CreateTopic	acs:mns:\$region:\$accountid:/topics/\$topicName
DeleteTopic	mns>DeleteTopic	acs:mns:\$region:\$accountid:/topics/\$topicName
SetTopicAttributes	mns:SetTopicAttributes	acs:mns:\$region:\$accountid:/topics/\$topicName
GetTopicAttributes	mns:GetTopicAttributes	acs:mns:\$region:\$accountid:/topics/\$topicName
ListSubscriptionByTopic	mns:ListSubscriptionByTopic	acs:mns:\$region:\$accountid:/topics/\$topicName/subscriptions
Subscribe	mns:Subscribe	acs:mns:\$region:\$accountid:/topics/\$topicName/subscriptions/\$subscriptionName
Unsubscribe	mns:Unsubscribe	acs:mns:\$region:\$accountid:/topics/\$topicName/subscriptions/\$subscriptionName
SetSubscriptionAttributes	mns:SetSubscriptionAttributes	acs:mns:\$region:\$accountid:/topics/\$topicName/subscriptions/\$subscriptionName
GetSubscriptionAttributes	mns:GetSubscriptionAttributes	acs:mns:\$region:\$accountid:/topics/\$topicName/subscriptions/\$subscriptionName
PublishMessage	mns:PublishMessage	acs:mns:\$region:\$accountid:/topics/\$topicName/messages

队列使用帮助

队列使用帮助

创建队列

首先使用阿里云 MNS 创建一个队列，在以下的示例中命名为 **MyQueue**。

1. 单击 **创建队列** 进入创建队列的对话框；

在对话框中填写队列属性，单击 **确定**，队列创建成功。

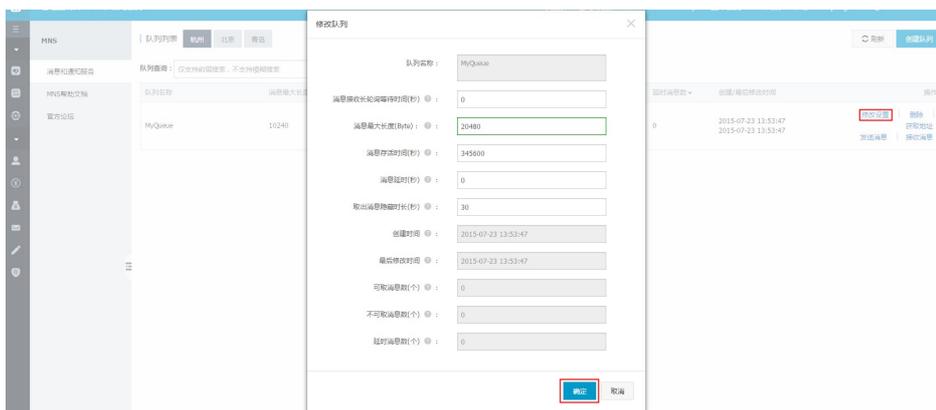


设置队列属性

确定 **MyQueue** 创建成功后，对 **MyQueue** 的属性进行设置。

1. 单击 **MyQueue** 右侧操作列表中的 **修改设置** 进入修改队列页面；

在弹出的对话框中，修改对应的属性，单击 **确认** 即可修改队列属性。

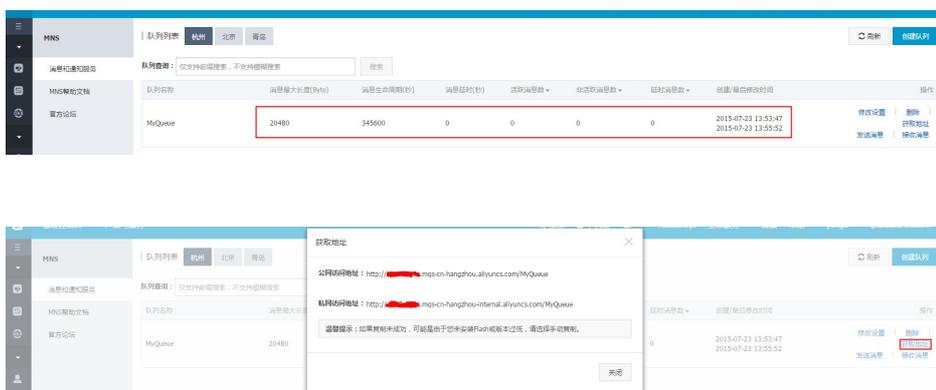


获取队列属性

确定 MyQueue 创建成功后，获取 MyQueue 的队列属性。

控制台默认显示队列基本属性，直接查看即可；

单击 MyQueue 右侧操作列表中的 **获取地址** 可查看队列的公网地址和私网地址。

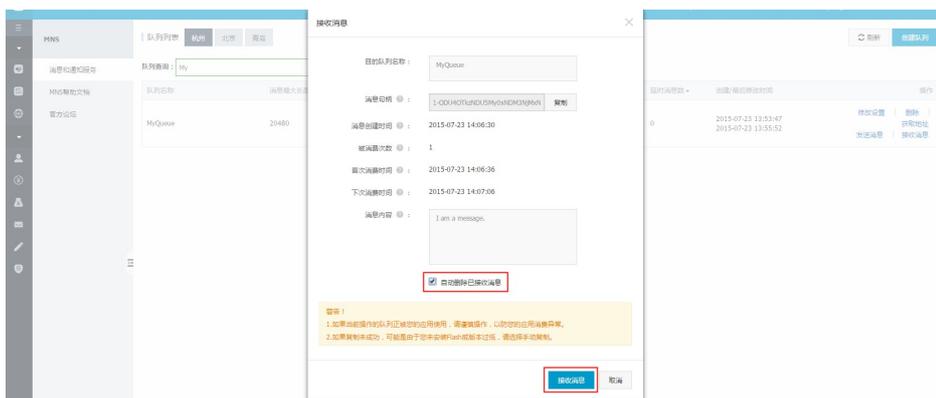


删除队列

确定 MyQueue 创建成功后，删除 MyQueue。

单击 MyQueue 右侧的 **删除** 进入删除页面；

单击 **删除** 即可删除队列。



获取队列列表

列出Account下以特定前缀开头的所有队列。

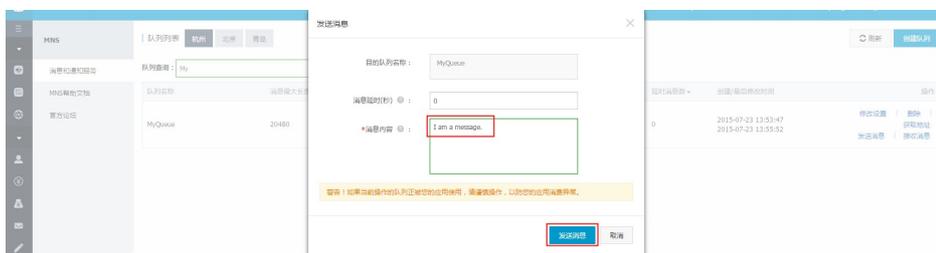
在队列查询的编辑框中输入对应的前缀字符串后单击 **搜索**，控制台列出对应前缀的队列。



发送消息

确定 MyQueue 创建成功后，向 MyQueue 发送消息。

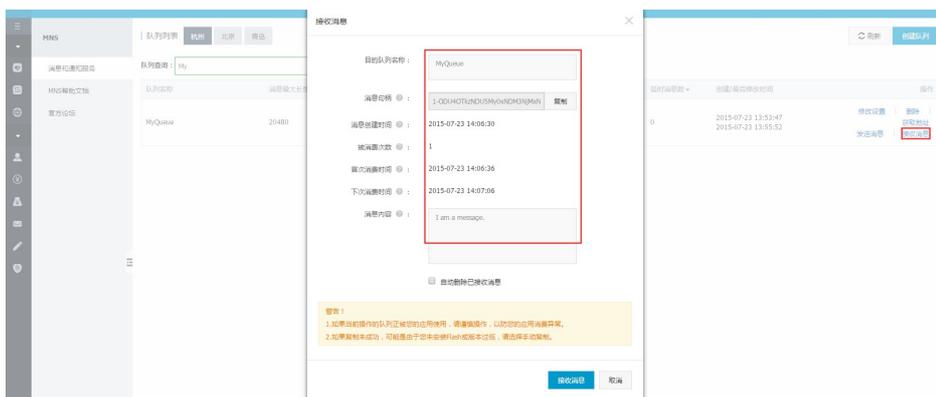
单击 MyQueue 右侧的 **发送消息**，在弹出对话框中，输入要发送的消息，单击 **发送消息**，消息发送成功。



消费消息

确定 MyQueue 创建成功后，从 MyQueue 中取出message，跟我们发送到 MyQueue 的消息进行对比，确定拿到的消息跟我们发送的消息一致。

单击 MyQueue 右侧的 **接收消息**，弹出的对话框中显示接收消息的相关属性。



删除消息

验证消息正确后，删除刚才取出来的 message。

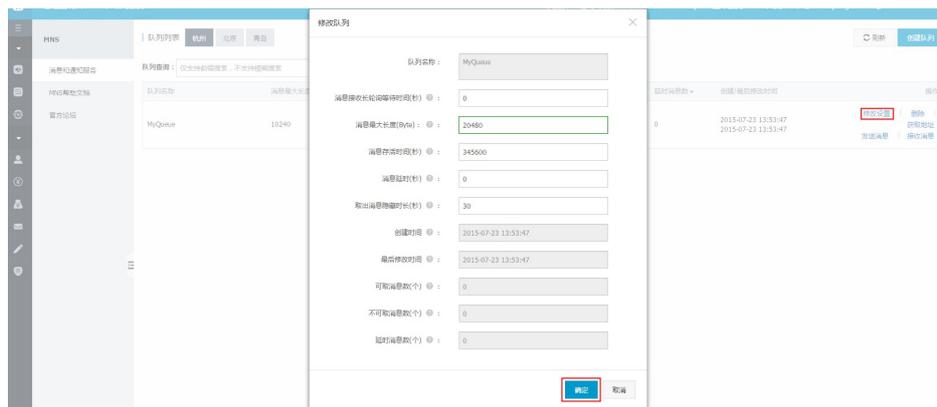
在接收消息的对话框中，选中 **自动删除已接收消息**，单击 **接收消息** 会删除本条消息，同时获取下一条消息。



设置队列属性

确定 MyQueue 创建成功后，对 MyQueue 的属性进行设置。

单击 MyQueue 右侧操作列表中的 **修改设置** 进入修改队列页面；
在弹出的对话框中，修改对应的属性，单击 **确认** 即可修改队列属性。



获取队列属性

确定 MyQueue 创建成功后，获取 MyQueue 的队列属性。

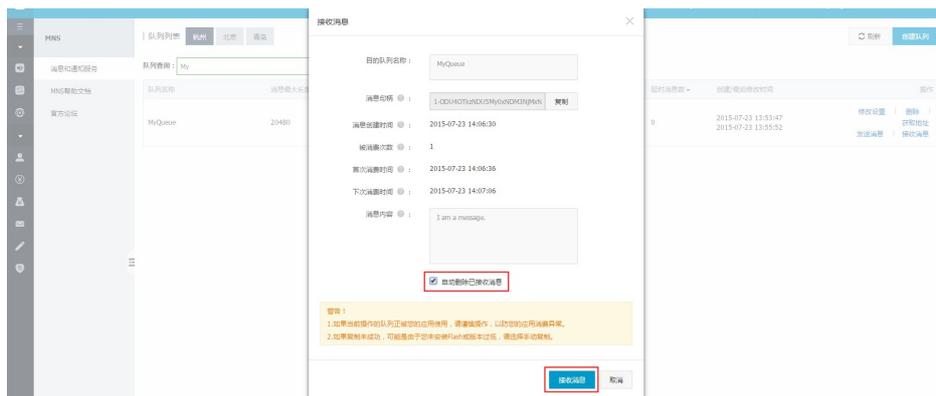
控制台默认显示队列基本属性，直接查看即可；
单击 MyQueue 右侧操作列表中的 **获取地址** 可查看队列的公网地址和私网地址。



删除队列

确定 MyQueue 创建成功后，删除 MyQueue。

单击 MyQueue 右侧的 **删除** 进入删除页面；
单击 **删除** 即可删除队列。



获取队列列表

列出Account下以特定前缀开头的所有队列。

在队列查询的编辑框中输入对应的前缀字符串后单击 **搜索**，控制台列出对应前缀的队列。



发送消息

确定 **MyQueue** 创建成功后，向 **MyQueue** 发送消息。

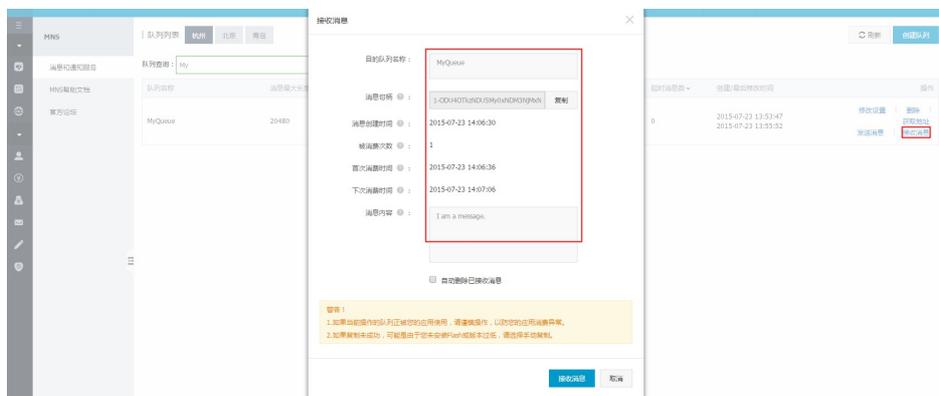
单击 **MyQueue** 右侧的 **发送消息**，在弹出对话框中，输入要发送的消息，单击 **发送消息**，消息发送成功。



消费消息

确定 **MyQueue** 创建成功后，从 **MyQueue** 中取出message，跟我们发送到 **MyQueue** 的消息进行对比，确定拿到的消息跟我们发送的消息一致。

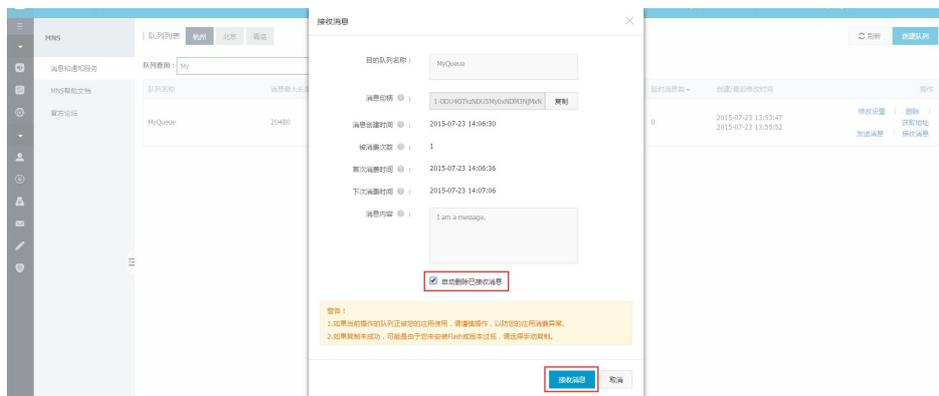
单击 **MyQueue** 右侧的 **接收消息**，弹出的对话框中显示接收消息的相关属性。



删除消息

验证消息正确后，删除刚才取出来的message。

在接收消息的对话框中，选中 **自动删除已接收消息**，单击 **接收消息** 会删除本条消息，同时获取下一条消息。



主题使用帮助

创建主题

创建一个名叫 **MyTopic** 的主题，主题创建成功后能在主题列表中查看到该主题。

单击左侧栏的 **发布订阅** 进入主题列表页面；

单击右上角的 **创建主题** 进入创建主题的对话框；

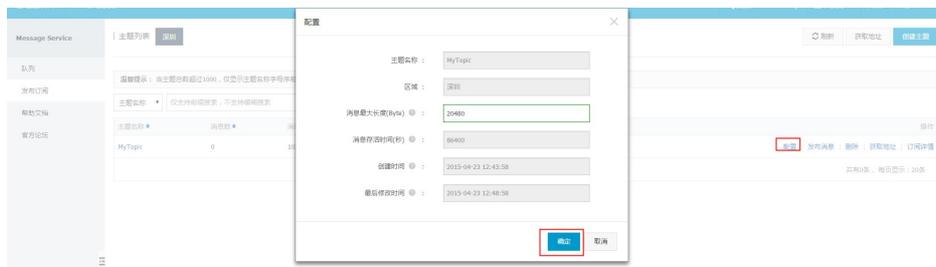
在对话框中输入属性值，单击 **确认** 即可创建主题；



设置主题属性

设置 MyTopic 的属性

- 单击 **MyTopic** 右侧操作列表的 **配置** 进入配置页面；
- 修改对话框中的属性值，单击 **确定** 即可修改主题属性；

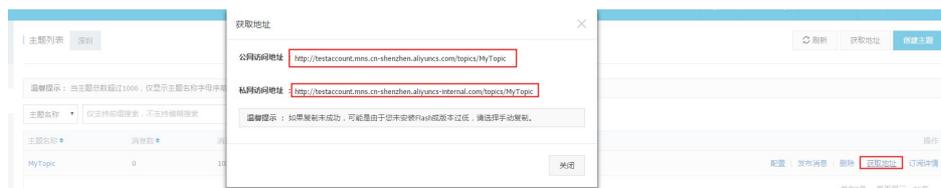
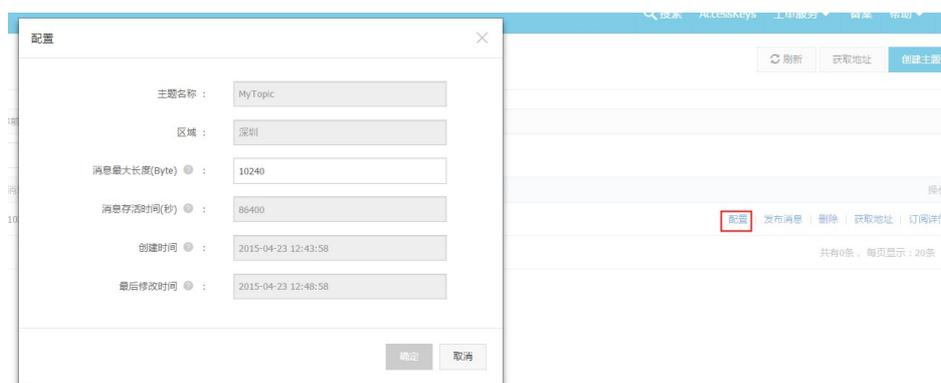


获取主题属性

获取 **MyTopic** 的相关属性。

- 在主题列表页面可查看到主题关键属性；
- 单击 **MyTopic** 右侧操作列表的 **配置** 进入配置页面可查看主题的全部属性；
- 单击 **MyTopic** 右侧操作列表的 **获取地址** 进入获取地址页面，可查看主题的公网地址和私网地址；





删除主题

单击 **MyTopic** 右侧操作列表的 **删除** 进入删除页面；

单击 **删除** 按钮即可删除主题；



获取主题列表

单击左侧栏的 **发布订阅** 进入主题列表页即可查看主题列表，以分页的方式展示；

如果希望查看以某前缀开始的主题列表，在 **主题名称** 后面的编辑框中输入对应前缀，此处是 **My**，单击右侧 **搜索** 按钮，即可查看主题名称以 **My** 为前缀的主题列表；

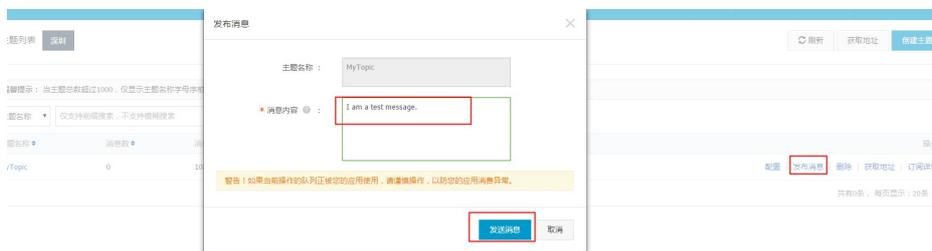


发布消息

发布一条消息到 MyTopic 中

单击 **MyTopic** 右侧操作列表中的 **发布消息**，进入发布消息页面；

在发布消息页面输入具体的消息内容，单击 **发送消息** 即可；



获取主题的订阅列表

查看 MyTopic 的订阅列表

单击 **MyTopic** 右侧操作列表的 **订阅详情** 进入订阅列表页，即可查看 **MyTopic** 的订阅列表，以分页的方式展示；

如果希望查看以某前缀开始的订阅列表，在 **订阅名称** 后面的编辑框中输入对应前缀，此处是 **My**，单击右侧 **搜索** 按钮，即可查看订阅名称以 **My** 为前缀的订阅列表；



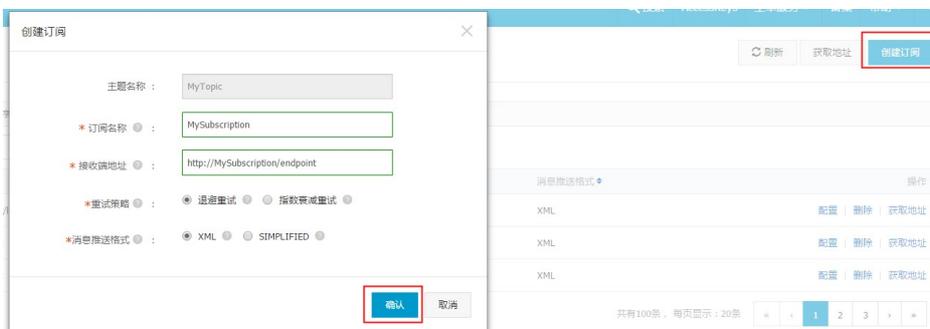
创建订阅

为 **MyTopic** 创建一个名为 **MySubscription** 的订阅，订阅创建成功后能在订阅列表中查看该订阅；

单击 **MyTopic** 右侧操作列表的 **订阅详情** 进入订阅列表页面；

单击右上角的 **创建订阅** 进入创建订阅页面；

在对话框中输入/选择订阅属性，单击 **确认** 即可；

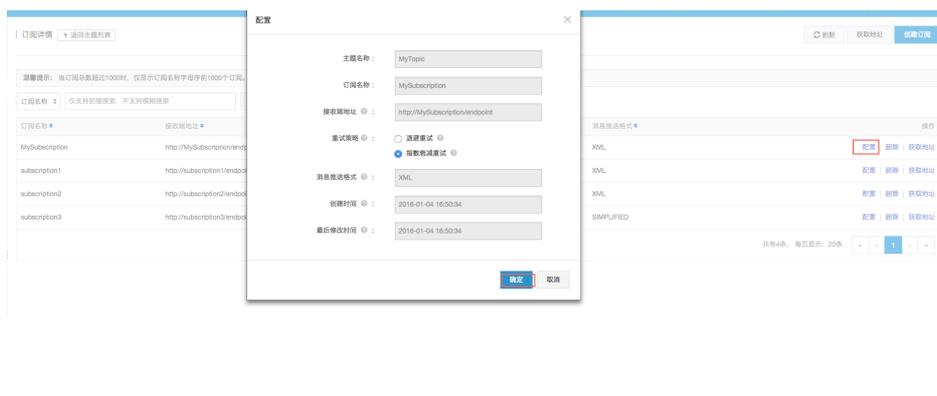


设置订阅属性

设置 MySubscription 的属性

单击 **MySubscription** 右侧操作列表的 **配置** 进入配置页面；

修改对话框中的属性值，单击 **确定** 即可修改订阅属性；



获取订阅属性

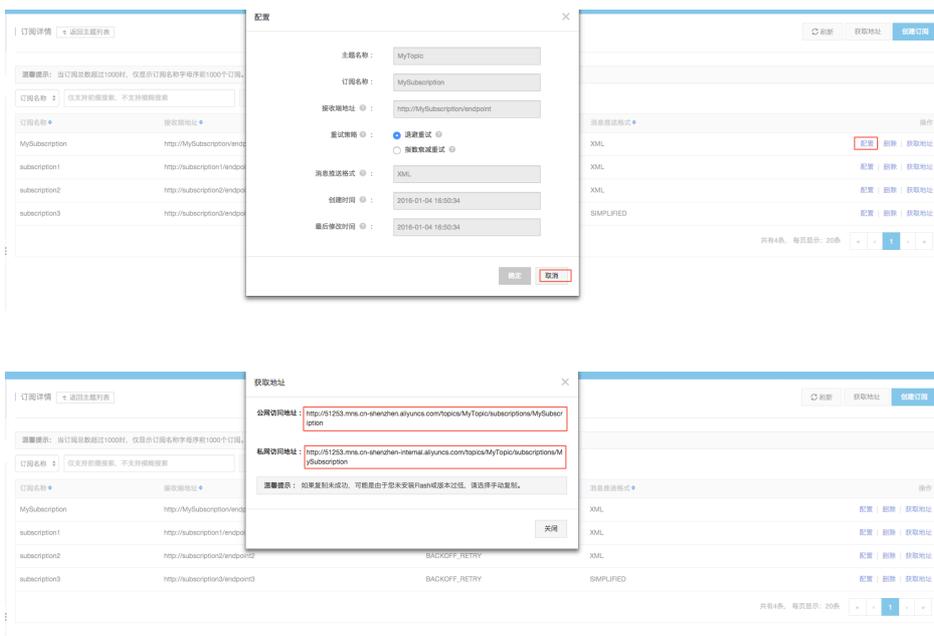
查看 MySubscription 的属性

在订阅列表页面可查看到订阅关键属性；

单击 **MySubscription** 右侧操作列表的 **配置** 进入配置页面可查看订阅的全部属性；

单击 **MySubscription** 右侧操作列表的 **获取地址** 进入获取地址页面，可查看订阅的公网地址和私网地址；

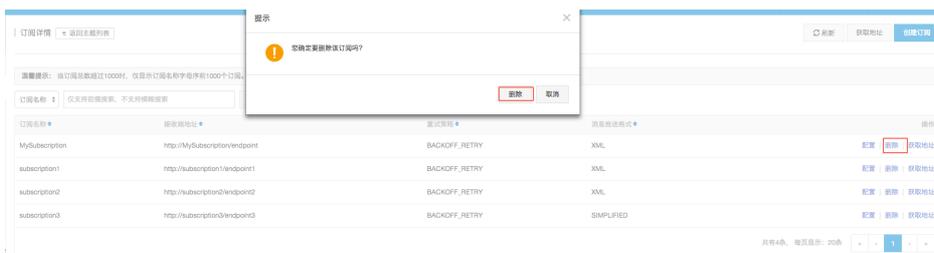




删除订阅

单击 **MySubscription** 右侧操作列表的 **删除** 进入删除页面；

单击 **删除** 即可删除订阅；



事件通知使用帮助

五步玩转事件通知

事件通知概述，请参考[介绍](#)

示例：

某个用户创建了一个Bucket：event-notification-test，然后他关心有哪些Object被上传到这个Bucket中了，他想把事件通知消息推送到Queue中，然后再慢慢消费，于是他就可以创建下面这样的事件通知规则：

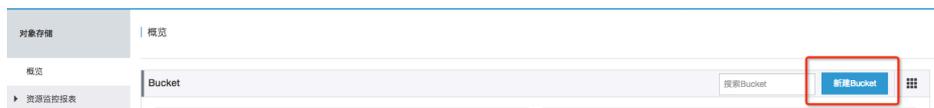
- 规则名称：event-notification-test-rule
- 资源描述：event-notification-test/
- 事件类型：PutObject、PostObject
- 接收终端：event-notification-test-queue

五步玩转事件通知：

- Step1. 创建Bucket
- Step2. 创建消息接收的Queue
- Step3. 创建事件通知规则
- Step4. 上传Object
- Step5. 查看Queue中是否接收到消息

Step1. 创建Bucket

- 登录阿里云官网，进入OSS控制台，点击“新建Bucket”，再依次填入BucketName：“event-notification-test”以及选择所属地域：“华南1”（此处方便测试可以设置读写权限为公共读写），如下：



新建Bucket

BucketName:

Bucket命名规范:
 » 1. 只能包含小写字母, 数字和短横线
 » 2. 必须以小写字母和数字开头和结尾
 » 3. bucketName的长度限制在3-63之间

所属地域:

读写权限:

私有: 对object的所有访问操作需要进行身份验证。
 公共读: 对object写操作需要进行身份验证; 可以对object进行匿名读。
 公共读写: 所有人都可以对object进行读写操作。

event-notification-test

存储区域: 华南 1 容量: 0.000KB 创建时间: 2016-07-05 17:22:25

存储空间 使用流量 每小时请求次数

资源监控延迟 暂无当日数据

COS域名

基础配置

Bucket权限: 公共读写 日志服务: 未开启 生命周期管理: 未开启

Step2. 创建消息接收的Queue

- 进入MNS控制台，选择“华南1”区域，点击“创建队列”，再依次输入队列名称：“event-notification-test-queue”，选择地域：“华南1”，其他按默认值即可，如下：

Message Service

队列列表 华北2 华东1 华北1 **华南1** 华东2 华南2 美国1

队列名称:

队列名称	消息生命周期	消息延时	活跃消息数	申请取消息数	延迟消息数	创建/准备修改时间	开启logging	操作
您还没有队列，立即创建								

新建队列
✕

* 队列名称 ? :

* 选择地域 :

消息接收长轮询等待时间(秒) ? :

取出消息隐藏时长(秒) ? :

消息最大长度(Byte) ? :

消息存活时间(秒) ? :

消息延时(秒) ? :

开启logging :

Message Service
队列列表 华北2 华东1 华北1 华南1 新加坡 美国1
刷新 帮助Support 创建队列

队列名称: 搜索

队列名称	消息生命周期(秒)	消息延迟(秒)	活跃消息数	非活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作
event-notification-test-queue	345900	0	0	0	0	2016-07-05 22:19:07 2016-07-05 22:19:07	false	修改配置 删除 获取地址 重置消息 接收消息

Step3. 创建事件通知规则

- 依然保持在MNS控制台，选择左边标签页：“事件通知”，点击“创建规则”，再依次输入规则名称：“event-notification-test-rule”、添加资源（点击“添加”按钮）、选择事件类型（可以多选）以及添加接收终端，如下图：
- 其中资源描述稍微复杂一些，可以参考资源描述后面的问号“？”，其中给了Tips，摘录如下（特别注意：**资源描述不能有交集**）：

资源描述可以是全名、前缀、后缀以及前后缀，不同资源描述不能有交集。

OSS资源包括Bucket和Object，通过“/”连接。

以Bucket (movie) 和Object (hello.avi) 为例。

1.全名：movie/hello.avi；

2.前缀：

movie/hello，表示movie中以hello为前缀的所有Object，

movie/，表示movie中所有的Object，

mov，表示以mov为前缀的所有Bucket中的所有Object；

3.后缀：.avi，表示所有Bucket中以.avi为后缀的Object；

4.前后缀：前movie/，后.avi，表示movie中所有以.avi为后缀的Object。

- 接收端目前支持HttpServer以及Queue两种，此处我们选择Step2.中创建的Queue。
- 创建一条新规则后，MNS会自动创建一个Topic，并且以指定的Endpoint创建订阅，可以从MNS控制台发布订阅的标签页进行查看，并且可以自由管理这些Topic与Subscription（比如创建新的订阅）。
- 新规则创建完成后，需要约10分钟的时间才能生效。



创建规则

温馨提示： 您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示： 任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区：

* 所在产品：

新建规则 默认规则

* 规则名称：

* 资源描述： 您还可以添加 5 个资源描述

* 事件类型：

* 接收终端： 您还可以添加 5 个接收终端

确认

取消

创建规则



温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区：

* 所在产品：

新建规则 默认规则

* 规则名称：

* 资源描述： 不能为空

注意：
资源描述
不能有交集

不能为空

不能为空

不能为空 -

您还可以添加 1 个资源描述

* 事件类型：

* 接收终端： http://开头的协议格式

队列名称

您还可以添加 3 个接收终端

确认

取消

创建规则



温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区：华南 1

* 所在产品：对象储存 OSS

新建规则 默认规则

* 规则名称：event-notification-test-rule

* 资源描述：前缀 event-notification-test/

添加 您还可以添加 4 个资源描述

* 事件类型：PostObject PutObject

* 接收终端：队列 event-notification-test-queue

添加 您还可以添加 4 个接收终端

确认

取消

规则名称	事件类型	资源	接收终端	主题名称	操作
event-notification-test-rule	PostObject, PutObject	前缀 event-notification-test/	event-notification-test-queue	mns-en-topics-oss-event-notifi...	删除

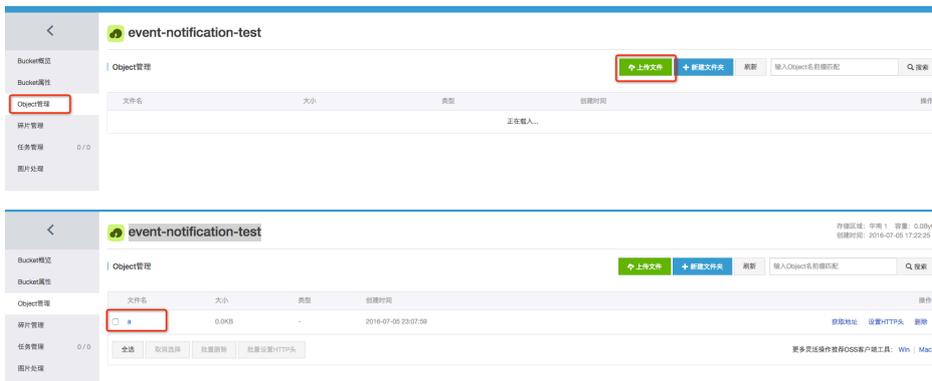
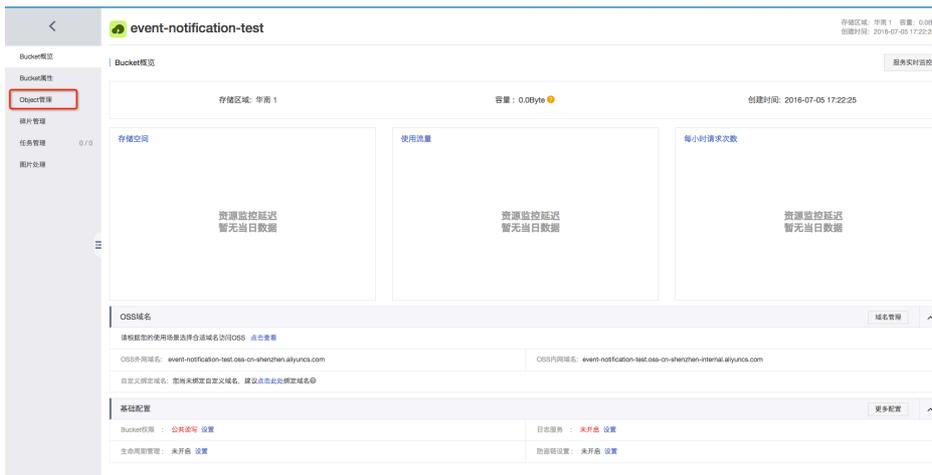
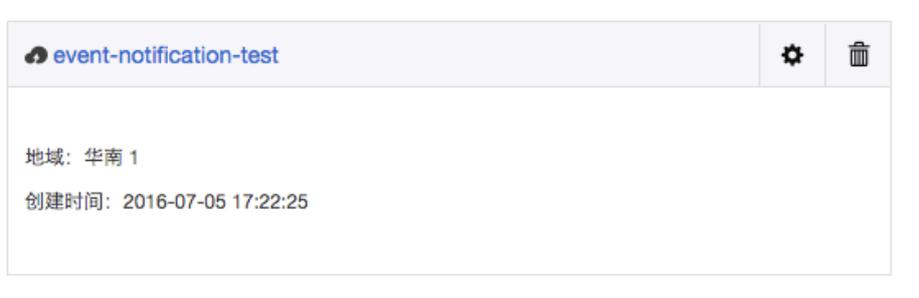
规则名称	事件类型	资源	接收终端	主题名称	操作
event-notification-test-rule	PostObject, PutObject	前缀 event-notification-test/	event-notification-test-queue	mns-en-topics-oss-event-notifi...	删除

主题名称	消息数	消息最大长度(Byte)	消息存活时间(秒)	开启Logging	操作
mns-en-topics-oss-event-notification-test-rule-146730405407430	0	65536	65400	false	配置 发布消息 删除 获取地址 订阅详情



Step4. 上传Object

- 新规则创建成功10分钟后，回到OSS控制台，尝试上传Object到Step1.创建的Bucket：“event-notification-test”中，验证是否能够触发事件通知。在OSS控制台选中“event-notification-test”，再点击左侧标签页的Object管理，然后在该标签页上传Object，如下图：



Step5. 查看Queue中是否接收到消息

- 上传Object成功后，回到MNS控制台，选中左侧标签页“队列”，并选中区域：“华南1”，查看

Step2.创建的队列中的活跃消息数是否为1，并可以点击“接收消息”，查看推送过来的具体事件通知消息。同时，在发布订阅标签页中，也可以看到Topic中多了一条消息，至此可验证事件通知成功。

如下图：

- 推送的事件通知消息内容如下，其具体含义可参考[Here](#)：

```
{
  "events": [
    {
      "eventName": "ObjectCreated:PutObject",
      "eventSource": "acs:oss",
      "eventTime": "2016-07-05T15:07:59.000Z",
      "eventVersion": "1.0",
      "oss": {
        "bucket": {
          "arn": "acs:oss:cn-shenzhen:12345:event-notification-test",
          "name": "event-notification-test",
          "ownerIdentity": "12345"
        },
        "object": {
          "deltaSize": 0,
          "eTag": "D41D8CD98F00B204E9800998ECF8427E",
          "key": "a",
          "size": 0
        },
        "ossSchemaVersion": "1.0",
        "ruleId": "event-notification-test-rule"
      },
      "region": "cn-shenzhen",
      "requestParameters": {
        "sourceIPAddress": "140.205.128.107",
        "responseElements": {
          "requestId": "577BCD4FF3D45CE477F2FA94",
          "userIdentity": {
            "principalId": "1390402650033793"
          }
        }
      }
    }
  ]
}
```

The screenshot shows the Message Service console interface. The 'Queue List' tab is active, displaying a table of queues. The queue 'event-notification-test-queue' is highlighted, showing 345600 total messages and 1 active message. The 'Active Messages' column is circled in red. The 'Queue Name' column is also circled in red. The 'Queue ID' column is circled in red. The 'Create/Last Update Time' column shows '2016-07-05 22:19:07'. The 'Logging' column shows 'false'. The 'Actions' column contains 'Modify Settings', 'Delete', and 'Receive Messages' (circled in red).

队列名称	消息生命周期数	消息队列数	活跃消息数	非活跃消息数	延迟消息数	创建/最后一次更新时间	开启Logging	操作
event-notification-test-queue	345600	0	1	0	0	2016-07-05 22:19:07 2016-07-05 22:19:07	false	修改设置 删除 接收消息

接收消息

✕

目的队列名称：

消息句柄：

消息创建时间：2016-07-05 23:08:00

被消费次数：1

首次消费时间：2016-07-05 23:11:58

下次消费时间：2016-07-05 23:12:28

消息内容：

```

{"events": [{"
  "eventName": "ObjectCreated:PutObject",
  "eventSource": "acs:oss",
  "eventTime": "2016-07-05T15:07:59.000"}]}

```

自动删除已接收消息

警告!

- 1.如果当前操作的队列正被您的应用使用，请谨慎操作，以防您的应用消费异常。
- 2.如果复制未成功，可能是由于您未安装Flash或版本过低，请选择手动复制。

接收消息

取消

主题列表	对比	订阅	订阅	消息	刷新	获取Endpoint	创建主题
<p>警告提示：当主题总数超过1000，仅显示主题名称字母序前1000个主题。您可以直接搜索主题名称查找对应主题，或单击DSOK获取完整主题列表。</p> <p>主题名称：<input type="text" value="仅支持前缀搜索，不支持模糊搜索"/> <input type="button" value="搜索"/></p>							
主题名称	消息数	消息最大长度(bytes)	消息存储时间(秒)	开启logging	操作		
msg-err-topics-oss-event-notification-test-rule-146772A0540143D	1	65536	86400	false	配置	发布消息	删除

默认规则事件通知

事件通知概述，请参考介绍

消息服务事件通知功能支持默认规则：不匹配用户其他规则的事件均会匹配默认规则。即：

- 如果用户仅有一条默认规则，则用户在某产品中（OSS）所有产生的事件均会匹配这条规则，对应的

消息通知会推送到默认规则描述的接收端中。

- 如果有多条其他规则与默认规则共存，则优先尝试匹配其他规则，都匹配的情况下，则会匹配默认规则，对应的消息通知会推送到默认规则描述的接收端中。

示例：创建默认规则，匹配所有OSS事件：

- Step1. 创建消息接收的Queue
- Step2. 创建默认事件通知规则
- Step3. 创建任意Bucket并上传Object
- Step4. 查看Queue中是否接收到消息

Step1. 创建消息接收的Queue

- 登录阿里云官网，进入MNS控制台，选择“华南1”区域，点击“创建队列”，再依次输入队列名称：“event-notification-default-queue”，选择地域：“华南1”，其他按默认值即可，如下：

The screenshot shows the '新建队列' (Create Queue) dialog in the MNS console. The '队列名称' (Queue Name) field is highlighted with a red box and contains the text 'event-notification-default-queue'. The '选择地域' (Select Region) dropdown is also highlighted with a red box and shows '华南 1'. Other fields like '消息接收长轮询等待时间(秒)', '取出消息隐藏时长(秒)', '消息最大长度(Byte)', '消息存活时间(秒)', and '消息延时(秒)' are empty. The '开启logging' (Enable logging) toggle is currently turned off. At the bottom right, there are '确认' (Confirm) and '取消' (Cancel) buttons.

Message Service | 队列列表 | 华北2 | 华东1 | 华北1 | 华东2 | 华南1 | 新加坡 | 美国1

队列

队列搜索： 仅支持前缀搜索，不支持模糊搜索 搜索

队列名称	消息生命周期	消息积压	活跃消息数	半活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作
event-notification-default-queue	346600	0	0	0	0	2016-07-20 09:30:48 2016-07-20 09:30:48	false	修改设置 删除 获取输出 接收消息

Step2. 创建默认事件通知规则

- 依然保持在MNS控制台，选择左边标签页：“事件通知”，点击“创建规则”，再选择默认规则，最后添加接收终端，如下图：
- 接收端目前支持HttpServer以及Queue两种，此处我们选择Step2.中创建的Queue。
- 创建一条新规则后，MNS会自动创建一个Topic，并且以指定的Endpoint创建订阅，可以从MNS控制台发布订阅的标签页进行查看，并且可以自由管理这些Topic与Subscription（比如创建新的订阅）。
- 新规则创建完成后，需要约10分钟的时间才能生效。

Message Service | 队列列表 | 华北2 | 华东1 | 华北1 | 华东2 | 华南1 | 新加坡 | 美国1

队列

队列搜索： 仅支持前缀搜索，不支持模糊搜索 搜索

队列名称	消息生命周期	消息积压	活跃消息数	半活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作
event-notification-default-queue	346600	0	0	0	0	2016-07-20 09:41:11 2016-07-20 09:41:11	false	修改设置 删除 获取输出 接收消息

Message Service | 事件通知列表 | 华南1

云产品通过MNS实现事件通知（异步回调），更多帮助。

规则名称	事件类型	资源	接收终端	主题名称	操作
<p>① 没有查询到符合条件的记录</p> <p>② 您还没有事件通知规则 立即创建</p>					

创建规则



温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区：

* 所在产品：

新建规则 默认规则

温馨提示：不匹配任何已有规则的事件都会匹配默认规则，请确认是否符合要求

* 规则名称：

* 资源描述：

* 事件类型：

* 接收终端： 您还可以添加 5 个接收终端

创建规则



温馨提示：您在同地区下的同产品中最多可以创建10条规则，新规则约10分钟后生效。

温馨提示：任意两个规则不能有交集，单条规则任意两个资源描述不能有交集

* 地区： 华南 1

* 所在产品： 对象储存 OSS

新建规则 默认规则

温馨提示：不匹配任何已有规则的事件都会匹配默认规则，请确认是否符合要求

* 规则名称： DefaultEnRule

* 资源描述： mns-en-reserved-default

* 事件类型： Default

* 接收终端： 队列 event-notification-default-queue

添加 您还可以添加 4 个接收终端

确认

取消

Message Service | 事件通知列表 华南 1

产品名称： 对象储存 OSS

规则名称	事件类型	资源	接收终端	主题名称	操作
DefaultEnRule	Default	全名:mns-en-reserved-default	event-notification-default-que...	mns-en-topics-oss-DefaultEnRule...	删除

Message Service | 事件通知列表 华南 1

产品名称： 对象储存 OSS

规则名称	事件类型	资源	接收终端	主题名称	操作
DefaultEnRule	Default	全名:mns-en-reserved-default	event-notification-default-que...	mns-en-topics-oss-DefaultEnRule...	删除

Message Service | 主题列表 华北 1 华南 1 华南 1 华南 1

温馨提示：当主题总数超过1000，仅显示主题名称字母序前1000个主题。您可以直接搜索主题名称查找对应主题，或者通过SDK获取完整主题列表。

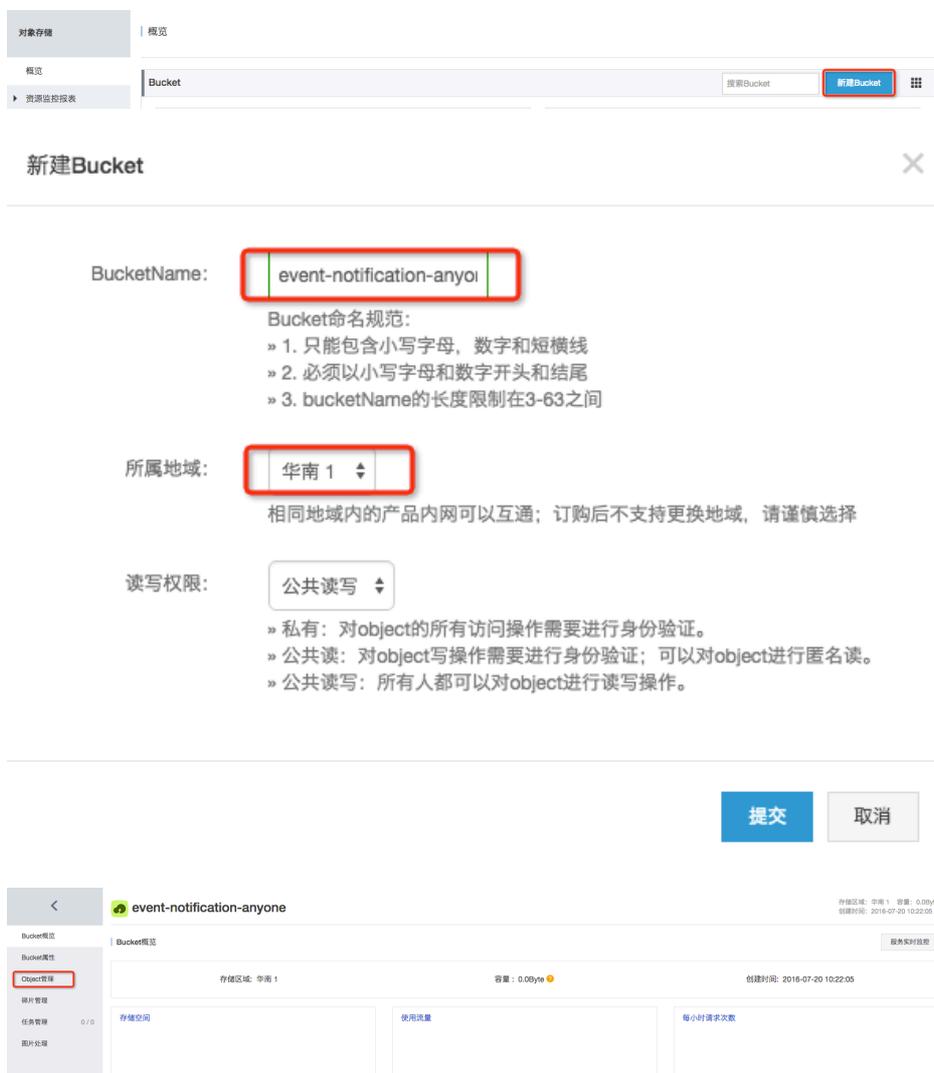
主题名称	消息数	消息最大长度(Byte)	消息方法(每秒)	开启logging	操作
mns-en-topics-oss-DefaultEnRule-1468981983177411	0	65536	86400	false	配置 发布消息 删除 获取地址 创建实例

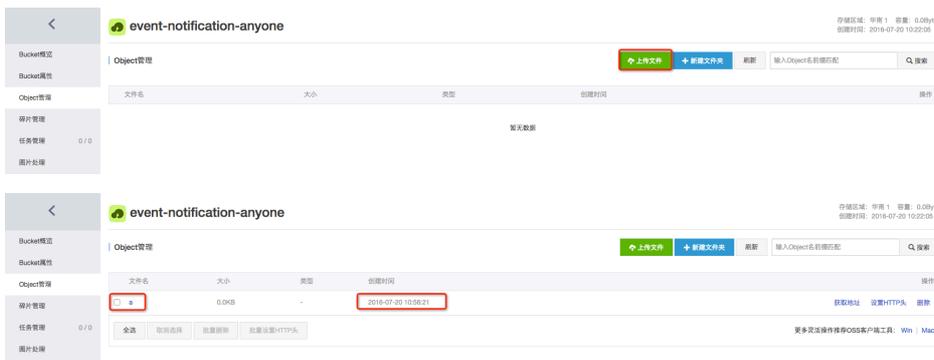
共有 1 条，每页显示： 20 条



Step3. 创建任意Bucket并上传Object

- 进入OSS控制台，点击“新建Bucket”，再依次填入BucketName：“event-notification-anyone”以及选择所属地域：“华南1”（此处方便测试可以设置读写权限为公共读写）
- 默认规则创建10分钟后，在刚刚创建的Bucket中随便上传一个Object，验证是否能触发事件通知。在OSS控制台选中“event-notification-test”，再点击左侧标签页的Object管理，然后在该标签页上传Object，如下图：





Step4. 查看Queue中是否接收到消息

- 上传Object成功后，回到MNS控制台，选中左侧标签页“队列”，并选中区域：“华南1”，查看Step1.创建的队列中的活跃消息数是否为1，并可以点击“接收消息”，查看推送过来的具体事件通知消息。
- 同时，在发布订阅标签页中，也可以看到Topic中多了一条消息，至此可验证事件通知成功。如下图：
- 推送的事件通知消息内容如下，其具体含义可参考Here：

```
{
  "events": [
    {
      "eventName": "ObjectCreated:PutObject",
      "eventSource": "acs:oss",
      "eventTime": "2016-07-20T02:56:21.000Z",
      "eventVersion": "1.0",
      "oss": {
        "bucket": {
          "arn": "acs:oss:cn-shenzhen:12345:event-notification-anyone",
          "name": "event-notification-anyone",
          "ownerIdentity": "12345",
          "object": {
            "deltaSize": 0,
            "eTag": "D41D8CD98F00B204E9800998ECF8427E",
            "key": "a",
            "size": 0,
            "ossSchemaVersion": "1.0",
            "ruleId": "DefaultEnRule",
            "region": "cn-shenzhen",
            "requestParameters": {
              "sourceIPAddress": "140.205.128.110",
              "responseElements": {
                "requestId": "578EE8558051B29CC3735797",
                "userIdentity": {
                  "principalId": "1390402650033793"
                }
              }
            }
          }
        }
      }
    }
  ]
}
```



接收消息



目的队列名称 : event-notification-default-queue

消息句柄 : 1-ODU4OTkzNDU5My0xNDY4OTg 复制

消息创建时间 : 2016-07-20 10:56:21

被消费次数 : 1

首次消费时间 : 2016-07-20 10:58:41

下次消费时间 : 2016-07-20 10:59:11

消息内容 : [{"events": [{"eventName": "ObjectCreated:PutObject", "eventSource": "acs:oss", "eventTime": "2016-07-20T02:56:21.000"}]}

自动删除已接收消息

警告!

- 1.如果当前操作的队列正被您的应用使用，请谨慎操作，以防您的应用消费异常。
- 2.如果复制未成功，可能是由于您未安装Flash或版本过低，请选择手动复制。

接收消息

取消

Message Service | 主题列表 | 消息 2 | 消息 1 | 消息 1 | 消息 1

队列

发布订阅

事件通知

日志管理

帮助文档

官方微博

消息服务: 当主题总数超过1000, 仅显示主题名称并带有1000个主题, 您可以通过主题名称查找主题, 或者通过SQS获取完整主题列表。

主题名称 * 仅支持普通主题, 不支持规则主题 搜索

主题名称 *	消息数 *	消息最大长度(Byte) *	消息存活时间(秒) *	开启Logging *	操作
msg-qn-topics-oss-DefaultRule-1466891983177411	1	65536	86400	false	配置 发布消息 删除 获取地址 订阅详情

共有1条, 每页显示: 20条

日志管理使用帮助

子账号授权准备

子账号授权准备

如果您使用子账号进行日志管理的配置，在开始配置之前，需要先给子账号授予相应的权限，这篇将——介绍如何授予这些权限。

在开始授权之前，请先使用主账号登录RAM的控制台。

1. 创建授权策略

1.1 列出RAM所有Role的权限

按图示操作，进入授权策略创建页，选择空白模板；



按照图示操作，修改策略名称和策略内容，策略名称可以自定义；



策略内容

```

{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ram:ListRoles",
      "Resource": "acs:ram:*:*:*"
    }
  ]
}

```

1.2 查看/设置MNS账号属性权限

按图示操作，进入授权策略创建页，选择空白模板；



按照图示操作，修改策略名称和策略内容，策略名称可以自定义；

创建授权策略
✕

STEP 1: 选择权限策略模板
STEP 2: 编辑权限并提交
STEP 3: 新建成功

* 授权策略名称: 1. 自定义权限名称
长度为1-128个字符, 允许英文字母、数字, 或“-”

备注: 2. 备注信息, 建议填写

策略内容:

```

1 {
2   "Version": "1",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action":
7         [
8           "mns:SetAccountAttributes",
9           "mns:GetAccountAttributes"
10        ],
11      "Resource": "acs:mns:*:*"
12    }
13  ]
14 }
```

3. 授权内容

[授权策略格式定义](#)
[授权策略常见问题](#)

4. 单击创建

上一步
新建授权策略
取消

策略内容

```

{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action":
        [
          "mns:SetAccountAttributes",
          "mns:GetAccountAttributes"
        ],
      "Resource": "acs:mns:*:*"
    }
  ]
}
```

1.3 LogService的ListProject和ListLogStore权限

按照图示, 新建一个空白的自定义授权策略;



按照图示，修改策略名称和策略内容，策略名称可以自定义；



策略内容

```

{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "log:List*",
      "Resource": "acs:log:*:*:*"
    }
  ]
}
    
```

1.4 OSS的ListBuckets权限

- 按照图示，新建一个空白的自定义授权策略；



- 按照图示，修改策略名称和策略内容，策略名称可以自定义；



策略内容

:

```

{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "oss:ListBuckets",
      "Resource": "acs:oss:*:*:*"
    }
  ]
}

```

2. 授权策略给予账号

2.1 按照图示，找到对应子账号，进入授权页面



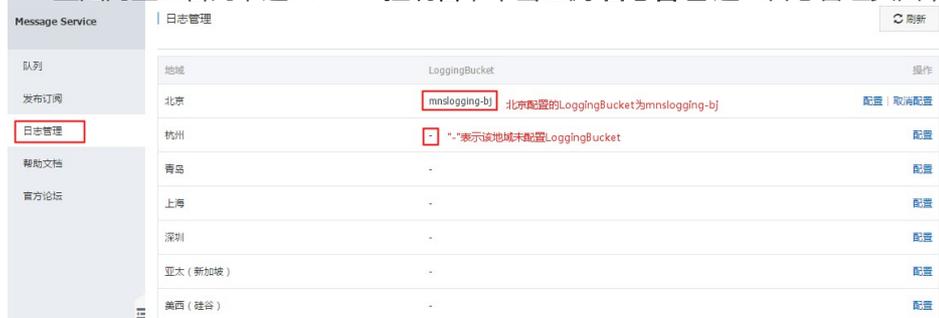
2.2 按照图示，将刚创建的策略授予子账号



推送日志到LogService

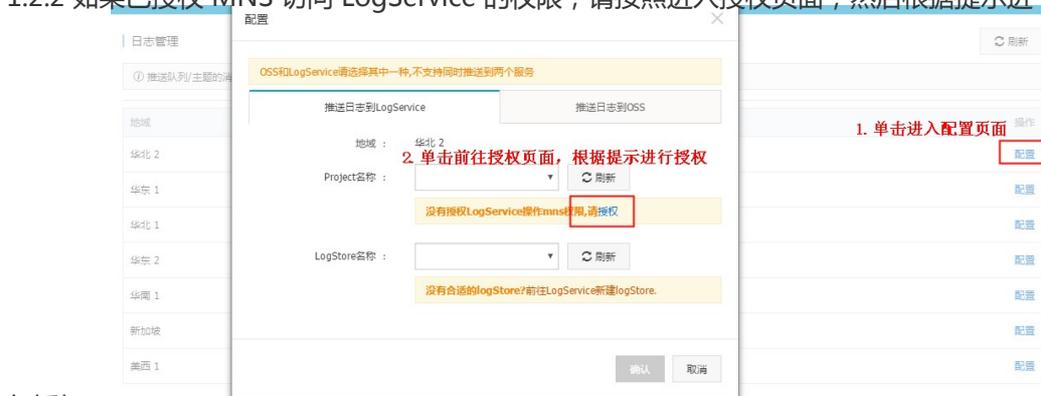
MNS 最新版本推出了日志管理功能，用户可以将日志推送到LogService中，然后在LogService查看消息轨迹等信息，该文件将介绍如何配置 MNS 的日志推送到LogService。

2.1 登陆阿里云官网，进入 MNS控制台，单击左侧 日志管理 进入日志管理页面；



1.2 单击杭州地域右侧的 **配置** 进入LoggingBucket配置页面；

- 1.2.1 如果已开通LogService服务，请直接到1.2.2步；如果未开通，请进入LogService产品页按照提示开始服务；
- 1.2.2 如果已授权 MNS 访问 LogService 的权限，请按照进入授权页面，然后根据提示进

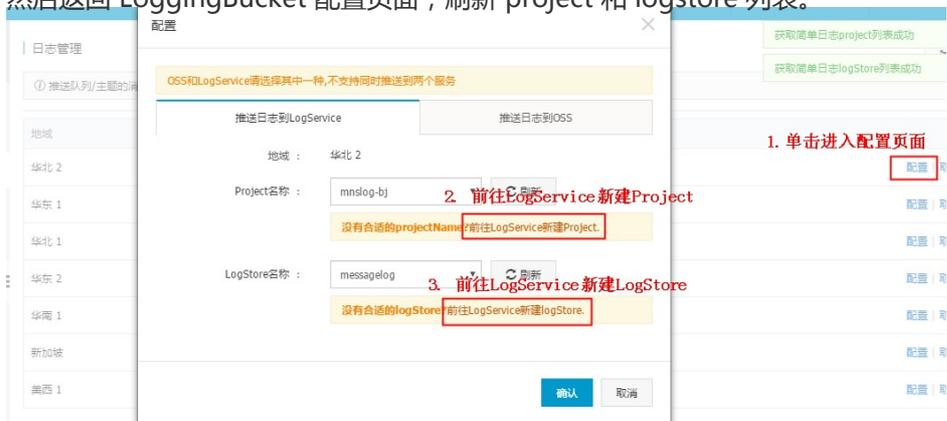


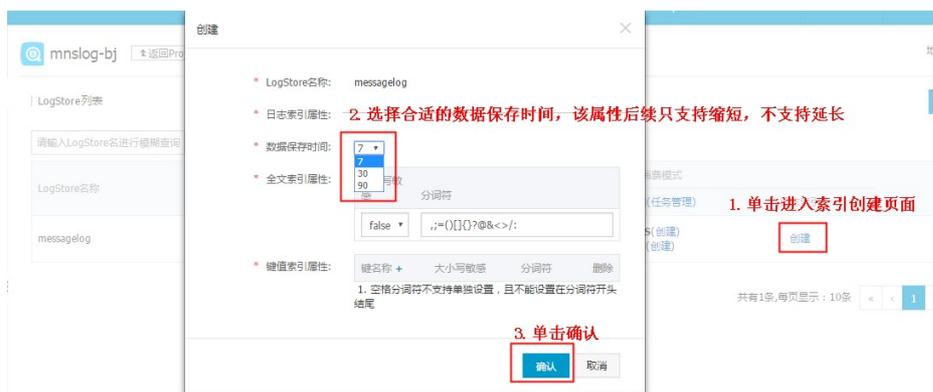
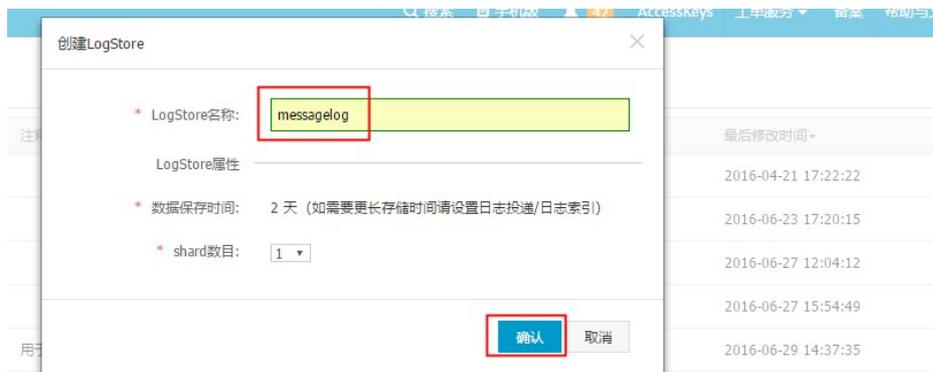
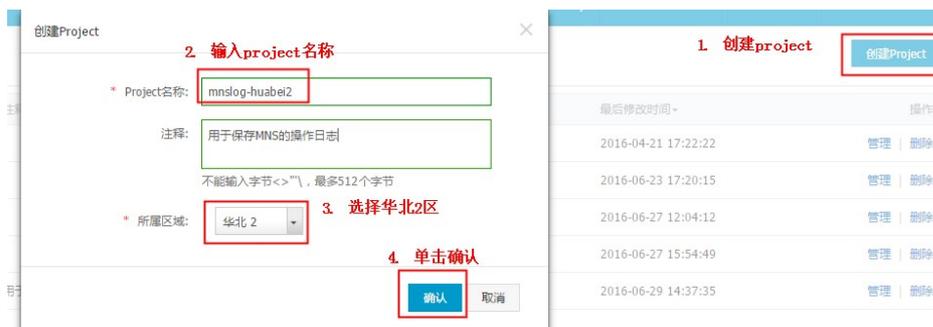
行授权；

- 1.2.3 如果没有合适的project或者logstore，请直接到1.2.4步；如果有合适的project和logstore，选择对应 project 和 logstore；



- 1.2.4 如果已执行1.2.3步，请跳过此步，这步介绍如何新建project和logstore；
 需要注意每个地域只能配置该地域的 project，因此在创建 project 的时候一定要选择对应地域（此处以华北2区为例）；
 创建project后，为该project创建LogStore；
 然后为LogStore创建索引，只有创建了索引才支持查询；
 然后返回 LoggingBucket 配置页面，刷新 project 和 logstore 列表。





- 1.2.5 完成配置后，可以在日志管理页面查看各个地域的LoggingBucket



下一步

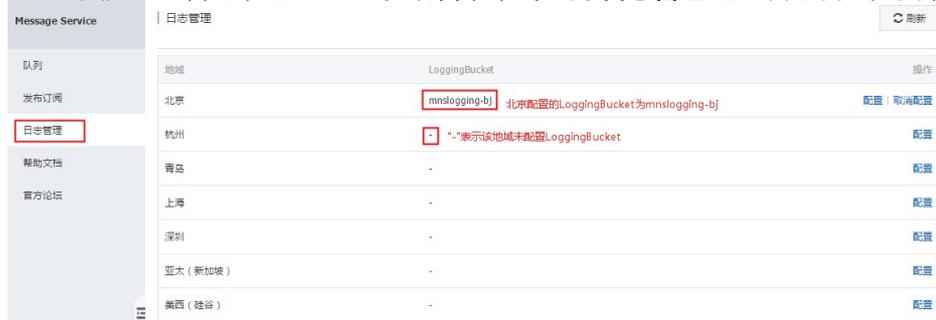
- 如何开启队列日志功能
- 如何开启主题日志功能
- 如何查看LogService中的日志

推送日志到OSS

MNS 最新版本推出了日志管理功能，用户可以将消息操作日志推送到OSS中，该文件将介绍如何配置 MNS 的日志推送到OSS。

1. 配置LoggingBucket

1.1 登陆阿里云官网，进入 MNS控制台，单击左侧 **日志管理** 进入日志管理页面；



1.2 单击杭州地域右侧的 **配置** 进入LoggingBucket配置页面；

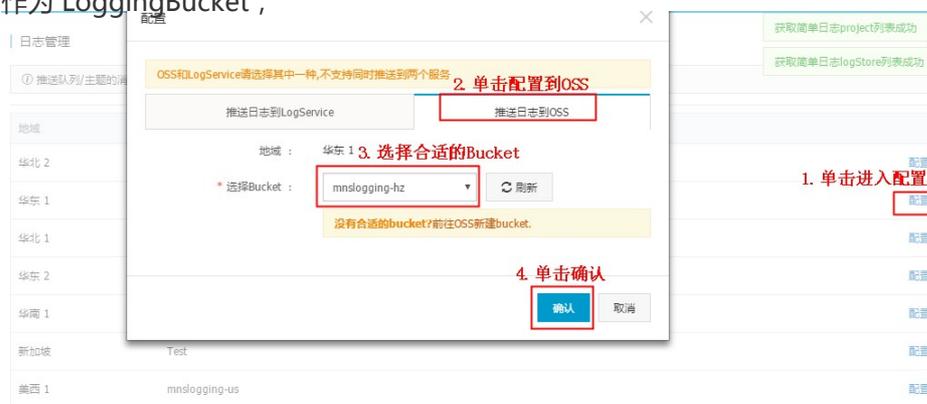
1.2.1 如果已开通OSS服务，请直接到1.2.2步；如果未开通，请单击 **立即开通** 按钮进行开通；



1.2.2 如果已授权 MNS 访问 OSS 的权限，请直接到1.2.3步；如果未授权，请单击 **前往RAM授权** 前往授权页面进行授权；



1.2.3 如果没有合适的Bucket，请直接到1.2.4步；如果有合适的Bucket，选择对应 Bucket 作为 LoggingBucket；



1.2.4 如果已执行1.2.3步，请跳过此步；新建 Bucket 作为 LoggingBucket；需要注意每个地域只能配置该地域的 Bucket，因此在创建 Bucket 的时候一定要选择对应地域（此处以杭州为例），然后返回 LoggingBucket 配置页面，刷新 Bucket 列表。



1.2.5 完成配置后，可以在日志管理页面查看各个地域的LoggingBucket

地域	LoggingBucket	操作
北京	-	配置
杭州	mnslogging-hz	配置 取消配置
青岛	-	配置
上海	-	配置
深圳	-	配置
亚太(新加坡)	-	配置
美国(硅谷)	-	配置

下一步

- 如何开启队列日志功能
- 如何开启主题日志功能
- 如何查看OSS中的日志

开启队列的日志功能

- 2.1 单击左侧 **队列** 进入队列管理页面；
- 2.2 单击队列右侧的 **修改设置** 按钮进入配置页面；
- 2.3 选择开启日志功能，单击 **确定** 完成配置；
- 2.4 完成配置后，返回队列列表页面，可以查看队列是否开启logging；



队列列表 北京 杭州 青岛 上海 深圳 亚太（新加坡） 美西（硅谷） 刷新 获取Endpoint 创建队列

队列查询： 搜索

队列名称	消息生命周期 (秒)	消息延时 (秒)	活跃消息数	非活跃消息数	延迟消息数	创建/最后修改时间	开启logging	操作
TestQueue	345600	0	0	0	0	2016-03-09 11:47:39 2016-03-09 11:56:55	返回队列列表页面，已开启logging true	修改设置 删除 获取地址 接收消息

开启主题的日志功能

- 3.1 单击左侧 **发布订阅** 进入主题管理页面；

- 3.2 单击主题右侧的 **配置** 按钮进入配置页面；
- 3.3 选择开启日志功能，单击 **确定** 完成配置；



查看日志--LogService

- MNS 推送日志到指定 LogStore 会延迟大约3分钟；
- 本文档仅介绍几种常用场景的查询，用户可以通过组合多个关键字来实现更加复杂的查询。

1. 进入日志查询页面

登陆LogService控制台，依次单击project和logstore的查询进入日志查询页面；

Project名称	注释	地域	创建时间	最后修改时间
mnslog		华南 1	2016-04-21 17:22:22	2016-04-21 17:22:22
mnslog1		华东 1	2016-06-23 17:20:15	2016-06-23 17:20:15
mnslog-qingdao		华北 1	2016-06-27 12:04:12	2016-06-27 12:04:12
mnslog-bj		华北 2	2016-06-27 15:54:49	2016-06-27 15:54:49
mnslog-hz-1	用于保存华东1区操作MNS的日志	华东 1	2016-06-29 14:37:35	2016-06-29 14:37:35



2. 查看队列消息的消息轨迹

- 在搜索框中输入队列名称和messageid
- 选择合适的时间范围后，单击搜索按钮即可查看该消息的详细操作日志；
- 格式：\$queueName and \$messageid
- 示例：MyQueue-160627-130907 and A86C6C2A7D060AB8-2-1559B08F15E-200000007



2. 查看队列消息写入量

- 在搜索框中输入队列名称和写入操作
- 选择合适的时间范围后，单击搜索按钮即可查看该队列消息写入量；
- 格式：\$queueName and (SendMessage or BatchSendMessage)
- 示例：MyQueue-160627-130907 and (SendMessage or BatchSendMessage)



3. 查看队列消息消费量

- 在搜索框中输入队列名称和消费操作
- 选择合适的时间范围后，单击搜索按钮即可查看该队列消息消费量；
- 格式：\$queueName and (ReceiveMessage or BatchReceiveMessage)
- 示例：MyQueue-160627-130907 and (ReceiveMessage or BatchReceiveMessage)



4. 查看队列消息删除量

- 在搜索框中输入队列名称和删除操作
- 选择合适的时间范围后，单击搜索按钮即可查看该队列消息删除量；
- 格式：\$queueName and (DeleteMessage or BatchDeleteMessage)
- 示例：MyQueue-160627-130907 and (DeleteMessage or BatchDeleteMessage)



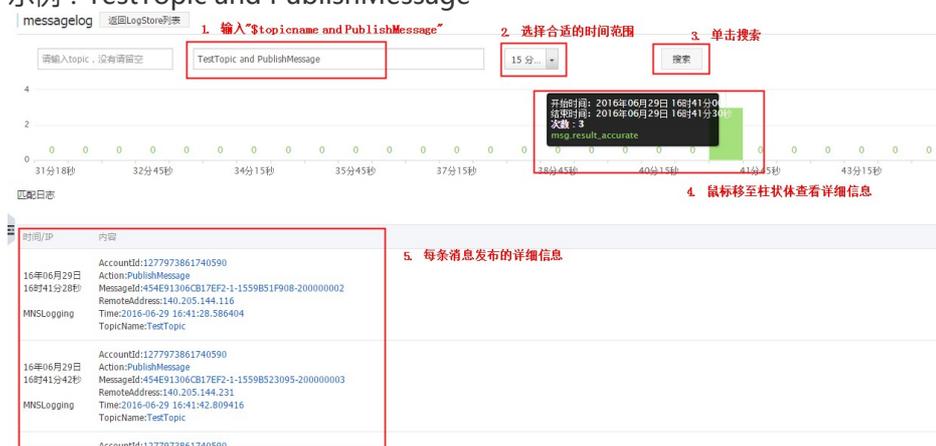
5. 查看主题消息的消息轨迹

- 在搜索框中输入主题名称和messageid；
- 选择合适的时间范围后，单击搜索按钮即可查看该主题消息的消息轨迹；
- 格式：\$topicName and \$messageid
- 示例：TestTopic and 454E91306CB17EF2-1-1559B51F908-200000002



6. 查看主题消息发布量

- 在搜索框中输入主题名称和发布操作
- 选择合适的时间范围后，单击搜索按钮即可查看该主题消息发布量；
- 格式：`$topicname and PublishMessage`；
- 示例：`TestTopic and PublishMessage`



7. 查看某个客户端消息处理量

- 在搜索框中输入客户端IP；
- 选择合适的时间范围后，单击搜索按钮即可查看该客户端所有的消息操作日志；
- 格式：`$ClientIP`
- 示例：`140.205.144.60`
- 如果希望查询某个客户端的某类操作日志，搜索框中增加具体操作即可，例如：`$ClientIP and (SendMessage or BatchSendMessage)`

messageLog 返回LogStore列表

1. 输入合适的ClientIP

2. 选择合适的时间范围

3. 单击搜索

请输入Topic, 没有请留空

140.205.144.60

自定义

选择时间范围

2016-06-28 15:20:00 - 2016-06-29 23:59:59

搜索

匹配日志

时间/IP	内容
16年06月28日 15时20分27秒	AccountId:1277973861740590 Action:ReceiveMessage MessageId:485ABBE692E5C087-1-15595DDCFE1-200000003 NextVisibleTime:1467098477 QueueName:MyQueue-160627-130907
16年06月28日 15时20分39秒	MNSLogging ReceiveHandleResponse:1CD140TLCNDUSNS0xNDY3MDk4NDc3LTERQA-- RemoteAddress:140.205.144.60 Time:2016-06-28 15:20:27.993817
16年06月28日 15时20分39秒	AccountId:1277973861740590 Action:ReceiveMessage MessageId:A86C6C2A7D060A8B-2-15595E15602-200000005 NextVisibleTime:1467098489 QueueName:MyQueue-160627-130907

查看日志--OSS

- 消息的操作日志以分钟为单位，按照固定的命名规则，生成一个 Object 写入指定的Bucket中。
- 队列和主题消息操作日志的Object路径不同，具体路径稍后介绍。
- 日志文件以 json 格式保存，用户可以直接下载文件进行处理。
- 整个日志文件是一个 map，其中 key 是 MessageId，value 是操作该消息的多条日志，每条日志包含多个字段，例如：Action, Time 等。
- MNS 推送日志到用户的 Bucket 会延迟大约15分钟。

队列消息的操作日志

<LoggingBucket>/QueueMessageLog/<QueueName>/YYYYmmdd/HHMMSS/log

示例：mnslogging/QueueMessageLog/TestQueue/20160229/101000/log

示例含义：该文件保存队列TestQueue从2016/02/29-10:10:00 到 2016/02/29-10:11:00的消息操作日志

- LoggingBucket 表示用户配置的 LoggingBucket 名称；
- QueueName 表示该 Object 所保存消息操作日志的队列名称；
- YYYY, mm, dd, HH, MM, SS 表示是该 Object 中日志的起始时间，分别是年，月，日，小时，分钟和秒；

主题消息的操作日志

<LoggingBucket>/TopicMessageLog/<TopicName>/YYYYmmdd/HHMMSS/log

示例：mnslogging/TopicMessageLog/TestTopic/20160229/100000/log

示例含义：该文件保存主题TestTopic从2016/02/19-10:00:00 到 2016/02/29-10:01:00的消息操作日志

- LoggingBucket 表示用户配置的 LoggingBucket 名称；

- TopicName 表示该 Object 所保存消息操作日志的主题名称；
- YYYY, mm, dd, HH, MM, SS 表示是该 Object 中日志的起始时间，分别是年，月，日，小时，分钟和秒；

2. 直接查看日志Object

- 1.1 登陆OSS控制台，单击配置的 LoggingBucket 进入Bucket主页面；



- 1.2 单击左侧的 **Object管理**，然后依次点击子目录即可查看相关的Object；



3. 使用工具查询日志

日志查询命令行工具提供queryqueuelog和querytopiclog命令，指定队列/主题名称、MessageId 和起止时间查询指定消息的操作日志。

具体的使用方法参考：日志查询命令行工具。