# 移动推送

# 操作指南

# 产品文档

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

# 文档目录


- 操作指南
  - 推送管理
    - 排查工具
    - 转化漏斗
    - 推送计划
    - 推送控制
    - 消息回撤
  - 推送高级功能
    - 标签功能
    - 多包名推送
    - 通知点击跳转
    - 定速推送
    - 通道策略
    - 消息折叠
    - 富媒体通知
    - 个性化通知
    - 账号推送


版权所有：腾讯云计算（北京）有限责任公司

# 操作指南
# 推送管理
# 排查工具

最近更新时间：2024-01-16 17:34:39

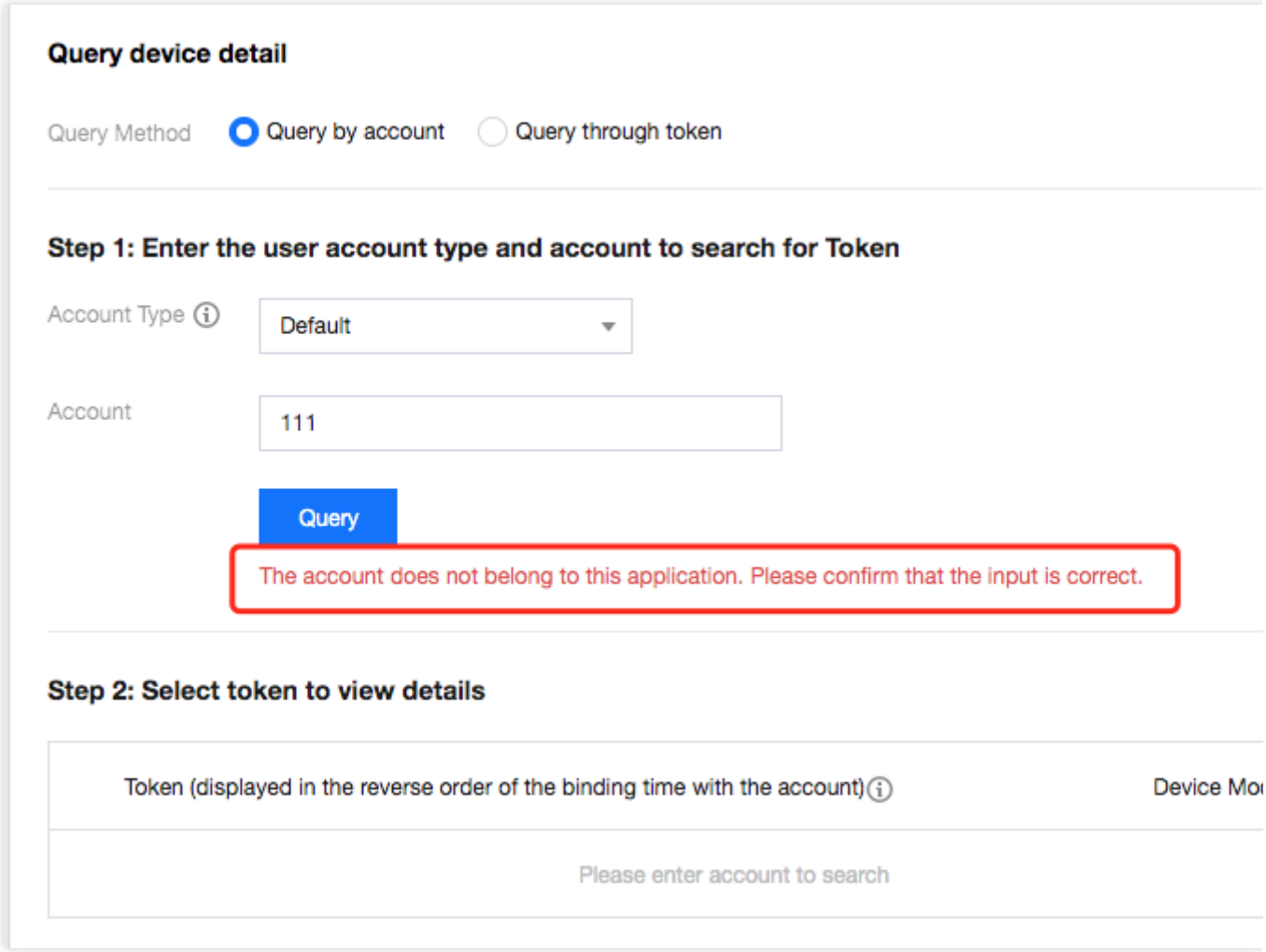
## 简介

当您在推送消息时出现异常情况（例如推送消息收不到或推送失败）时，可通过排查工具进行自助排查以及通过账号或者 Token 查询设备详情、设备绑定的账号以及设备绑定的标签。

## 常见场景

1. 开发人员在控制台推送一条消息给测试设备，控制台推送状态显示成功，但在手机通知栏未看到这条消息，通过 Token 查询设备详情发现通知栏权限关闭，在手机上开启通知栏权限后，再次触发注册，此时往这台设备发送消息，手机通知栏成功展示。
2. 运营人员通过用户账号推送消息，推送状态显示“推送失败”，在排查工具中输入用户账号及绑定的账号类型查询 Token，发现该应用下，此账号没有关联的设备 Token，可联系开发人员排查是否调用了 账号绑定（Android）或 账号绑定（iOS）接口。

版权所有：腾讯云计算（北京）有限责任公司

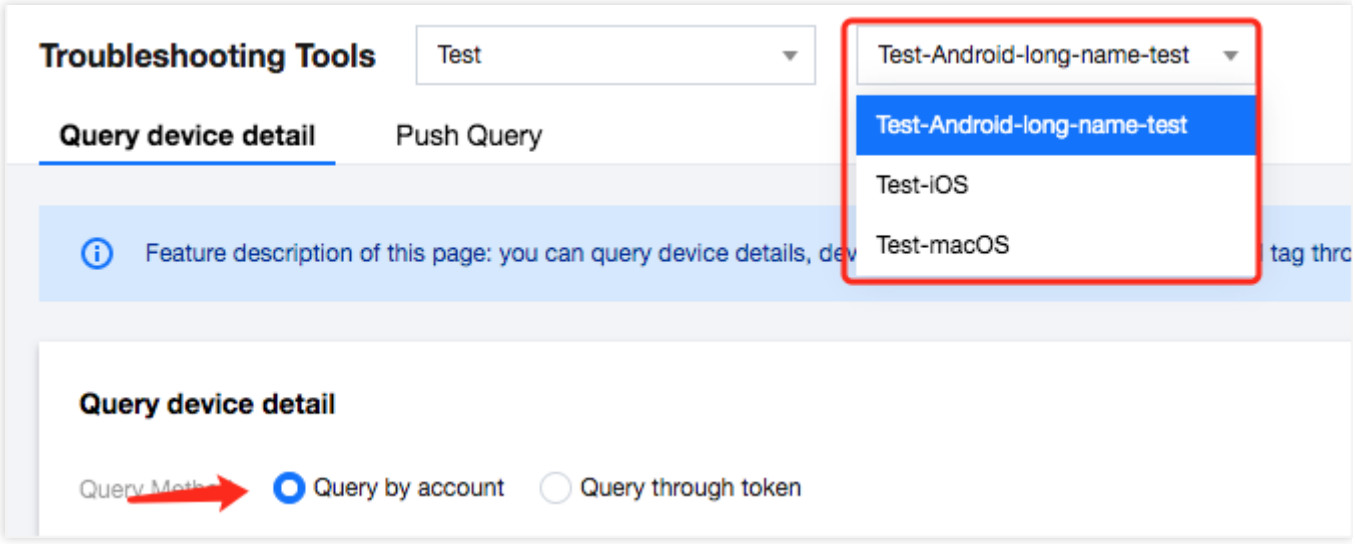



3. 开发人员完成了移动推送的接入工作后，对某一批设备 Token 推送消息，线上用户反馈没有收到该推送信息，获取该用户 Token 和 pushID 后，在推送排查得知，推送列表中并无该 Token。

# 操作步骤

### 通过用户账号查询

1. 登录 移动推送控制台。
2. 在左侧导航栏选择 **App推送管理** > **排查工具**。
3. 单击下拉框选择您需要查询产品及应用，选择**通过账号查询**。




4. 输入用户账号，单击**查询**。

5. 选择账号下关联的设备 Token 查看设备注册详情。

**说明：**

账号：与 Token 绑定的用户唯一 ID，一般为 OpenID、UID 等。

Token 列表展示规则：Token 按与账号绑定时间的倒序展示，本页面最多展示10个 Token，推送目标设置为账号推送时，系统默认推送给最近一次绑定该账号的设备，例如需要更改为推送给所有绑定该账号的设备，请在 控制台 或 API 推送时更改设置。

### 通过 Token 直接查询

1. 在排查工具页面，选择**通过 Token 直接查询**。

2. 输入设备 Token，单击**查询**，页面右侧会显示当前设备注册详情。

**说明：**

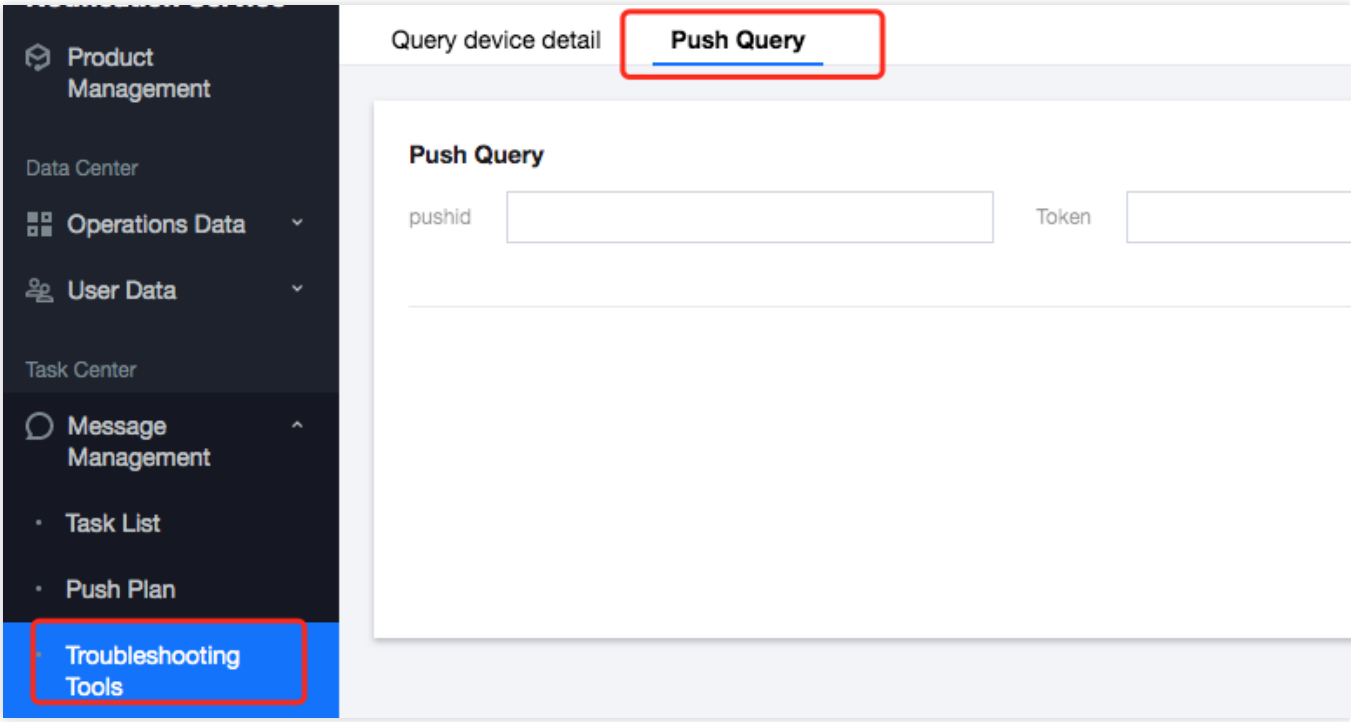
Token：移动推送分配给每一个设备的唯一ID，长度36位。

若您不知道如何获取设备 Token，可参考 SDK 集成文档中的 获取 Token 交互建议（Android）或 获取 Token 交互建议（iOS）。

### 推送查询

1. 在**排查工具**页面，单击**推送查询**。

版权所有：腾讯云计算（北京）有限责任公司

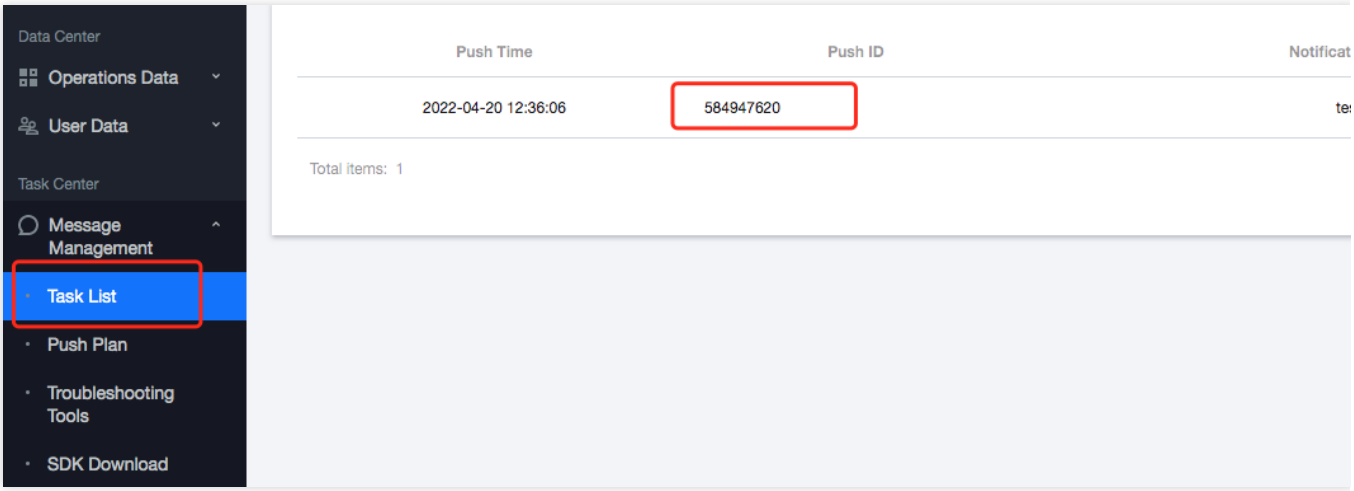



2. 输入 pushid（必选）、待查询的设备 Token（必选），单击**查询**，查看排查结果。

**说明：**

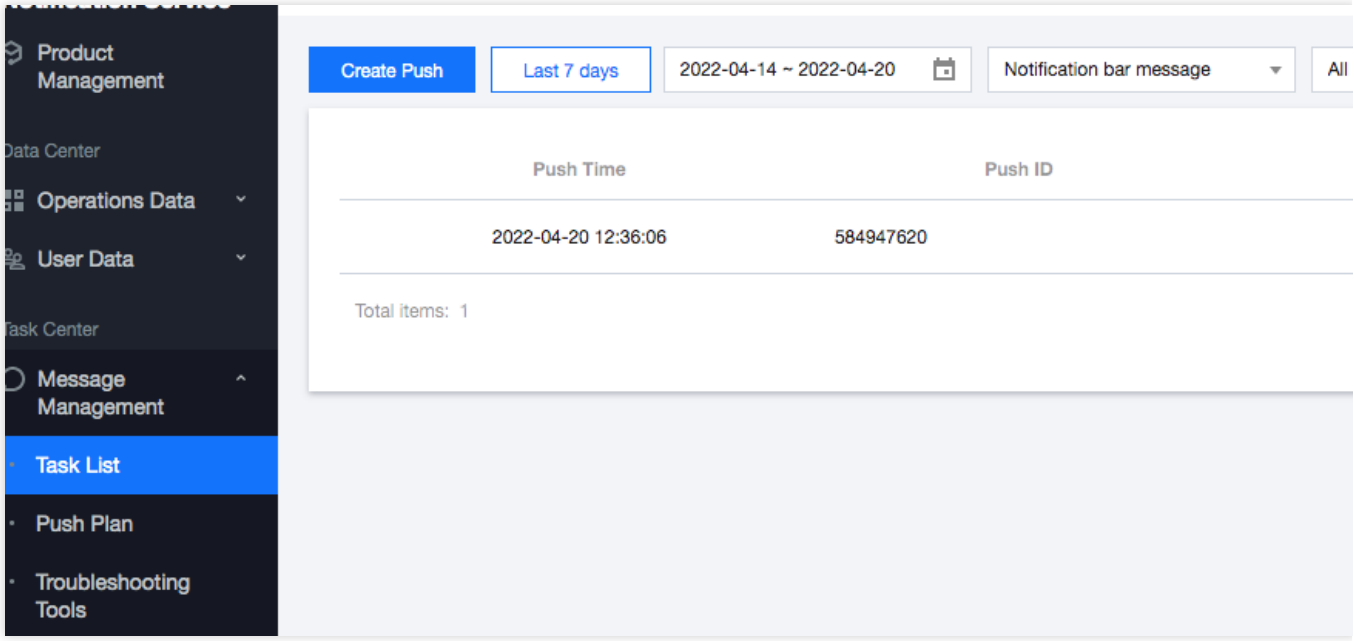
pushid 获取方式：

1. 可在左侧导航栏选择 **App 推送管理** > **推送任务**，获取需要查询的 PushID。




2. 在推送接口的应答参数中获取。

3. 在 **App 推送管理** > **推送任务**，更换 管理台创建 > API创建 及可查看到 API 推送数据




4. 若查询结果与实际情况不符或仍无法解决问题，可查看 推送常见问题 或联系 在线客服，并补充 pushID 和 Token 信息。

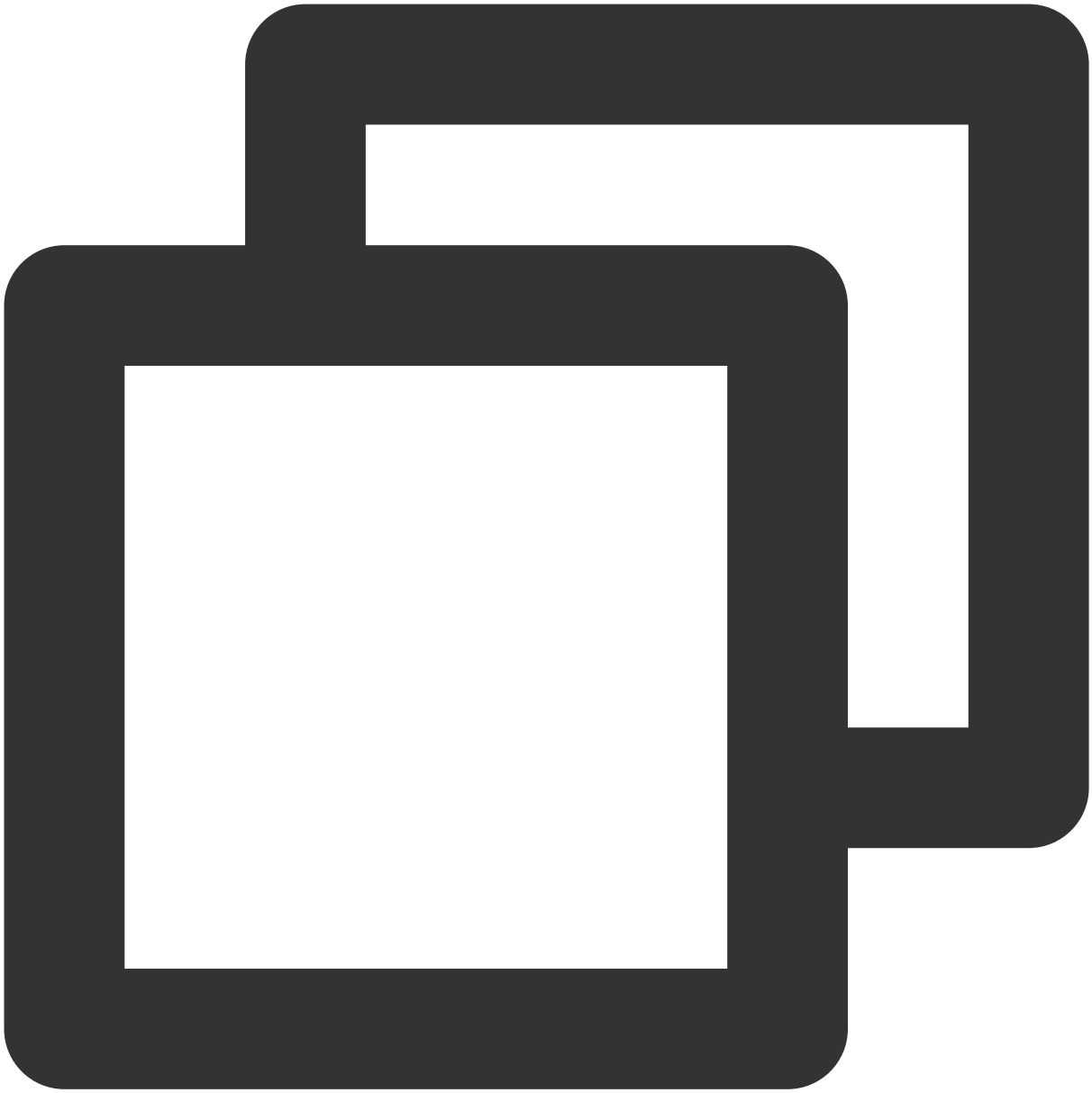
# 常见问题

### 设备查询常见问题

1. 设备 Token 什么情况下会失效？
Token 有效期为90天，若设备连续90天未连接移动推送服务器，则会被标记为无效设备。
若设备卸载了 App，当前 Token 会被标记为无效。
2. 通知栏权限已经开启，为什么设备查询显示关闭状态？
若手机通知栏设置从关闭到开启权限，需要在客户端重新触发一次注册请求，才可以把开启状态同步到移动推送服务器。

### 推送常见问题

1. 通过账号或者标签推送无法收到，但通过 Token 推送可以收到，是什么原因？
在设备注册详情页中查看该 Token 是否关联了推送的目标账号或者标签，若未关联则无法通过账号或标签推送。
2. 推送状态显示已完成，设备状态正常，为什么手机没有收到推送？
iOS 平台推送环境是否与 Token 注册环境一致，不一致则无法收到推送，推送环境选择说明详情可参考文档 推送环境选择。
检查 App 包名是否和**移动推送控制台>App推送管理>基础配置**中填写的包名一致，如果包名不一致检查是否开启 多包名推送功能。
安卓 P 及以上系统需要添加使用 Apache HTTP client 库，检查在 AndroidManifest 的 application 节点内是否有添加以下配置：



```
<uses-library android:name="org.apache.http.legacy" android:required="false"/>
```

小米手机确认是否被收纳到**不重要通知**。

魅族手机确认是否被收纳到**消息盒子**。

vivo 、OPPO 部分机型通知栏默认关闭，确认是否手动开启。

vivo 、华为部分机型，通知渠道的横幅、铃声等权限默认关闭，确认是否手动开启。

应尽量避免使用“test”、“测试”等字眼，部分厂商通道会拦截这类型的通知。

vivo、OPPO 需要厂商审核通过后才可以测试推送，请注意确认。

版权所有：腾讯云计算（北京）有限责任公司

# 转化漏斗

最近更新时间：2024-01-16 17:34:39

运营人员可在 转化漏斗 页面查看和分析应用在不同维度的推送转化数据。我们强烈建议运营人员每日例行访问转化漏斗页面，通过分析不同维度的数据转化效果，可从数据中发现促进转化的机遇与卡点，不断优化推送策略，将推送的价值最大化。

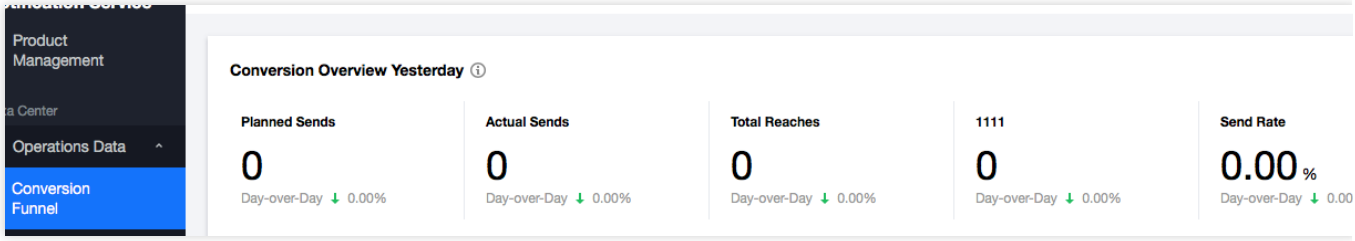
## 指标口径

**计划发送次数：**当日所有推送任务可下发的设备数加和，不去重
**实际发送次数：**当日所有推送任务实际下发到终端设备/厂商服务器的次数加和，不去重
**抵达次数：**当日所有推送任务**移动推送自建通道的抵达成功回执和厂商通道抵达成功回执**数量加和，不去重
**点击次数：**当日所有推送任务被点击的次数加和，不去重
**实发率（PV）：**实际发送次数/计划发送次数 * 100%
**抵达率（PV）：**抵达次数/实际发送次数 * 100%
**点击率（PV）：**点击次数/抵达次数 * 100%
**计划发送用户数：**当日所有推送任务可下发的去重设备数
**实际发送用户数：**当日所有推送任务实际下发的去重设备数
**抵达用户数：**当日所有推送任务抵达的去重设备数
**点击用户数：**当日所有点击推送的去重设备数
**实发率（UV）：**实际发送用户数 / 计划发送用户数 * 100%
**抵达率（UV）：**抵达用户数 / 实际发送用户数 * 100%
**点击率（UV）：**点击用户数 / 抵达用户数 * 100%
**说明：**
计划发送前过滤无效或者通知栏关闭的设备。
实际发送前过滤无法下发厂商通道且 App 离线的设备或其他。

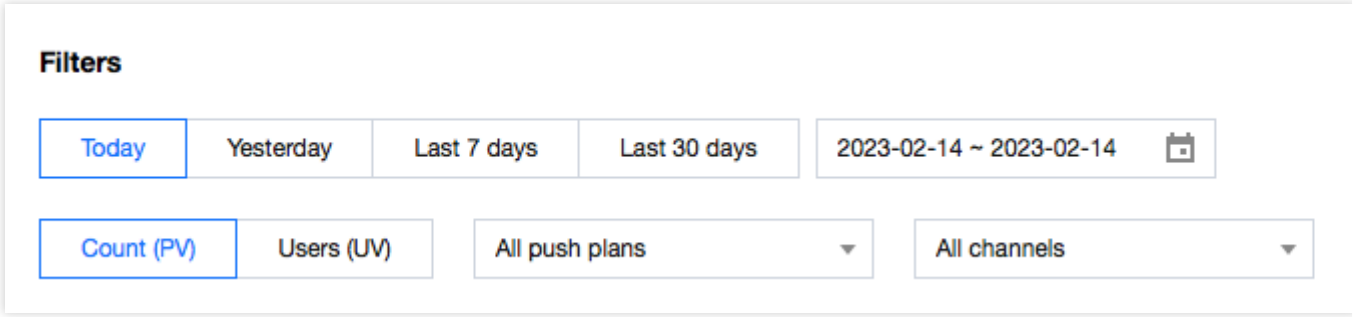
## 昨日转化总览

运营人员可在**昨日转化总览**模块快捷查看昨日整体转化效果，同时可通过日环比模块快捷洞察各项指标较前天的波动幅度。

版权所有：腾讯云计算（北京）有限责任公司




# 统计维度




1. 可通过**日期**筛选，查看不同时间区间内的数据转化趋势，支持查看近90天内的转化趋势。

**说明：**

若日期选择的起止日期相同（即选择**今天**、**昨天**、**2021-06-06 ~ 2021-06-06**）时，底部数据粒度为小时；起止日期非同一天时，数据粒度为天。

2. 可按**次数（pv）或用户（uv）**维度查看转化数据。若选择次数（pv），则数据不去重；若选择用户数（uv），则数据按设备维度去重。

例如：若某天对同一个设备发送了10条推送并成功抵达和点击，则次数（pv）维度点击数为10，而用户数（uv）维度点击数为1，其他指标同理。

3. 可按**推送计划**维度查看转化数据。我们强烈建议您将业务的不同场景划分为不同的 推送计划，然后分析不同场景推送的转化率。

例如：客户 A 的应用有**运营活动**、**促活专项**、**新用户礼包**三个推送计划，则该客户可通过筛选推送计划，查看不同场景的转化效果，进而持续优化增长策略。

4. 可按**推送通道**维度查看转化数据。可查看不同推送通道的转化效果。

5. 可按**消息类型**维度查看转化数据。若您同时使用移动推送发送通知栏消息、静默消息、透传消息等，可查看不同消息类型的转化效果。

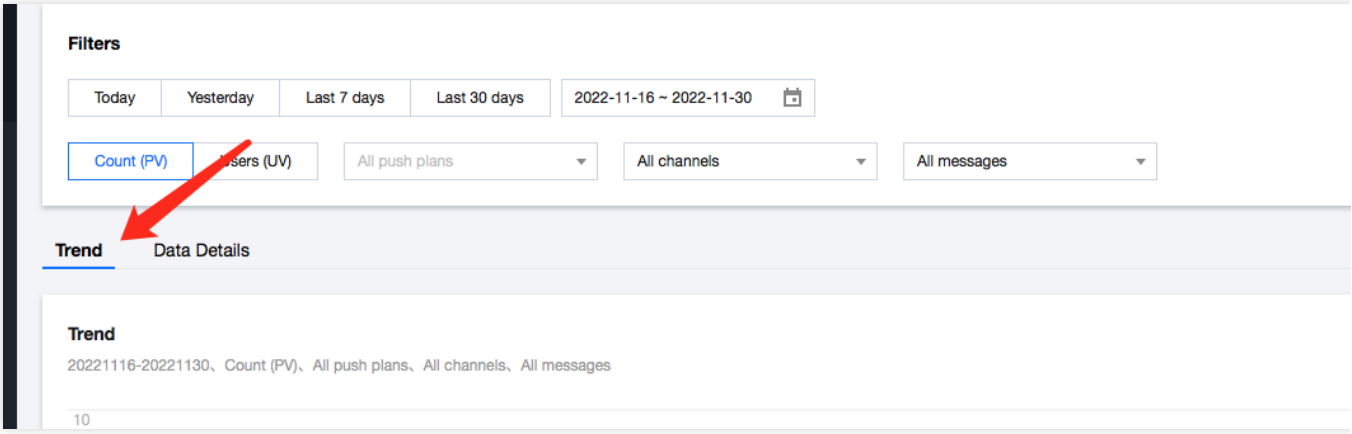
# 数据分析

版权所有：腾讯云计算（北京）有限责任公司

数据分析部分，分为数据趋势、数据明细、数据下载，跟随筛选条件动态变化。

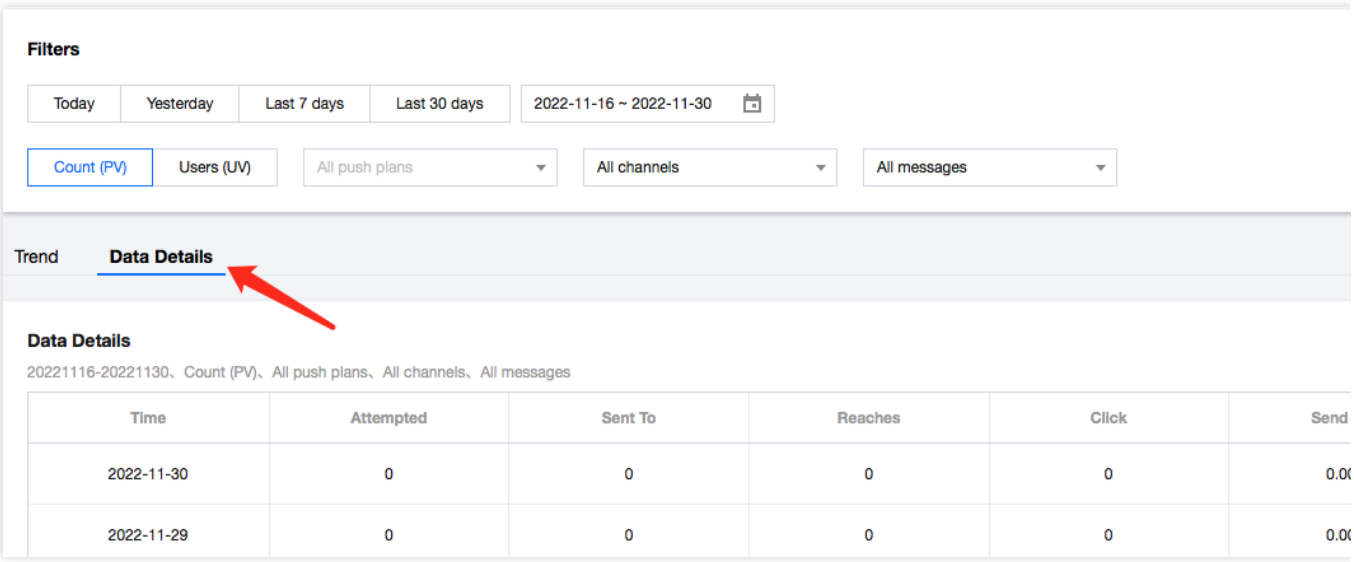
## 数据趋势

在数据趋势区域，您可以查看时间维度的推送转化趋势。




## 数据明细

1. 选择**数据明细**页签，可查看按筛选条件展示的详细数据。
2. 您可以单击**数据下载**导出 CSV 文件数据表，帮助您更详细的分析数据转化效果。




**说明：**

若统计周期跨天，则按照数据上报时间进行划归。即：若某条任务3月1日下发并抵达设备 A，但3月2日才被点击，则计划发送数、实际发送数、抵达数会被划归到3月1日，但点击数会被划归到3月2日。

版权所有：腾讯云计算（北京）有限责任公司

# 推送计划

最近更新时间：2024-01-16 17:34:39

## 简介

推送计划帮助运营或开发者以组合的方式管理不同推送目标的多条推送任务，同时按计划查看推送的整体效果，评估整体目标的达成情况。

## 应用场景

**管理不同运营目标的推送任务**

针对如618，双11等阶段性的活动，需要创建多条推送任务，或者新用户拉次日留存等有相同运营目标的多条推送任务，可以创建一个推送计划将其管理起来，按照计划的维度查看总消息触达量、用户点击量，从而查看最终的转化效果和营销目标达成情况。

**用户触发的通知和推送**

常见的有点赞、评论、分享、私信等不同类型，通过将每一类的所有单用户推送任务归类到同一计划，可以按天查看该类型的推送消息量、用户点击量等数据，评估该类型消息对于用户活跃的拉动效果。

## 操作步骤

### 控制台创建推送计划

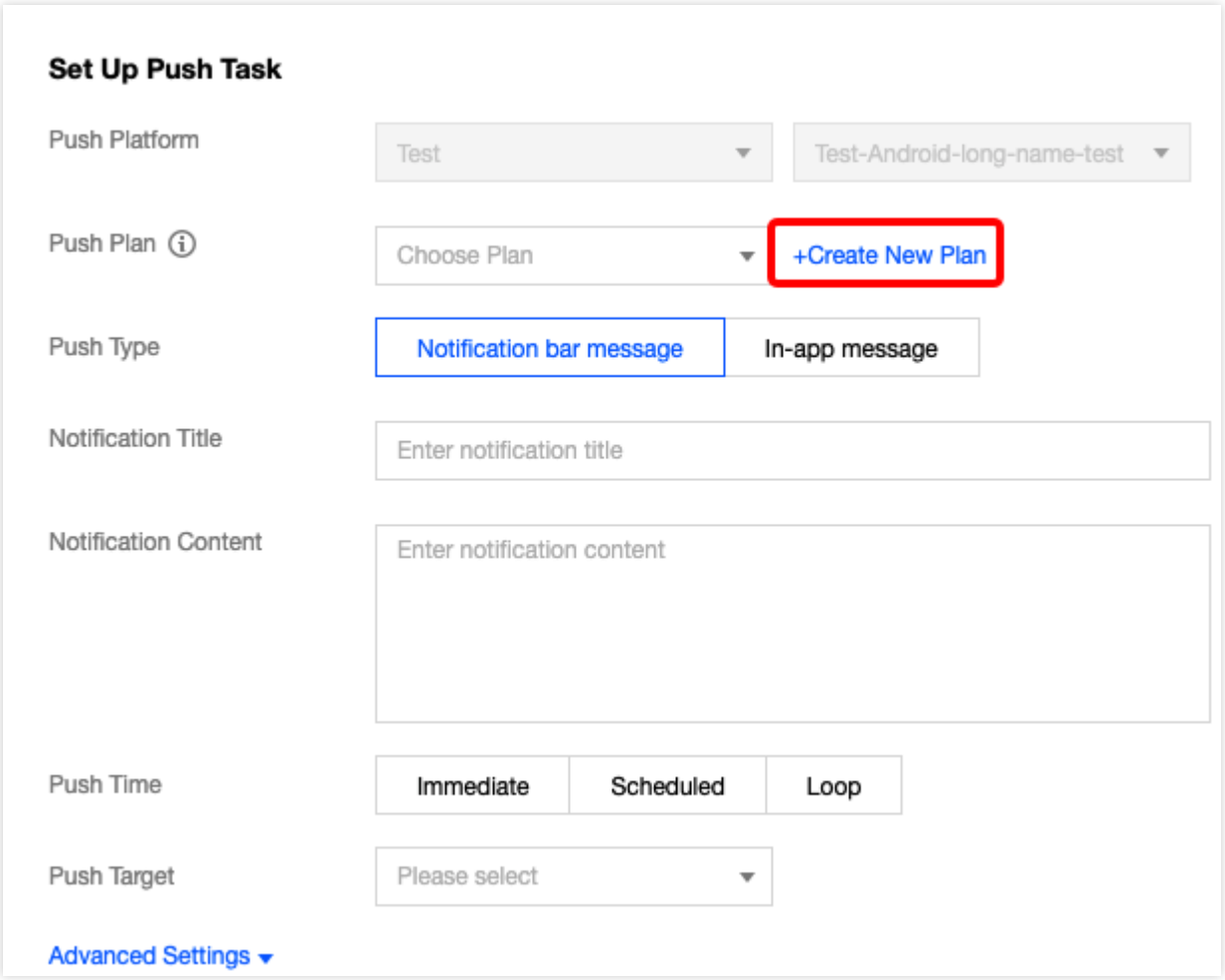
**方式一：**

1. 登录 腾讯移动推送控制台。
2. 在左侧导航栏中，选择 **App 推送管理** > **推送计划**，单击**新建计划**。
3. 在弹出的对话框中，输入计划名称和计划描述，单击**确定**。

创建完成后，可在推送计划页查看该计划的名称、描述、推送任务数；可随时编辑和删除该计划(默认计划除外)。

**方式二：**

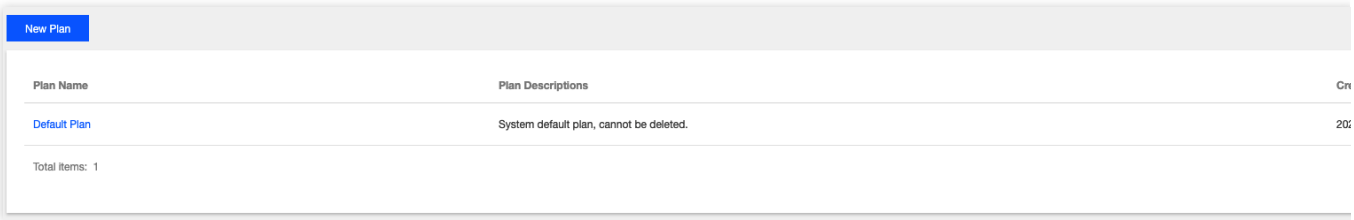
1. 登录 腾讯移动推送控制台。
2. 在左侧导航栏中，选择 **App 推送管理** > **推送任务**，单击**新建推送**。
3. 单击**新建推送计划**。




4. 在弹出的对话框中，输入计划名称和计划描述，单击**确定**。
创建完成后，本次推送自动选中该推送计划。

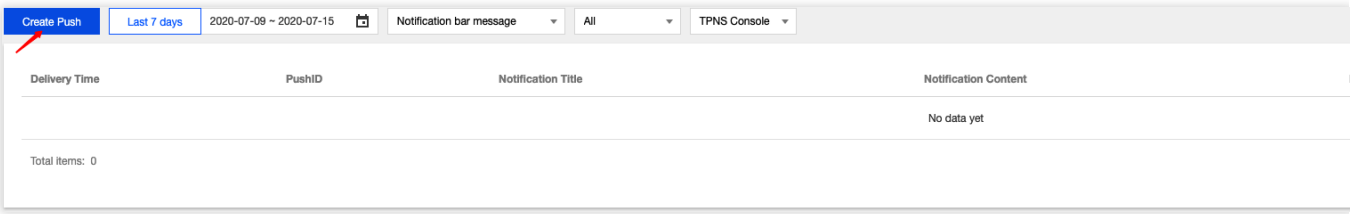
## 查看推送计划下的推送任务

1. 登录 腾讯移动推送控制台。
2. 在左侧导航栏中，选择 **App 推送管理** > **推送计划**，选中本次需要进行的推送计划，单击**详情**进入推送任务列表页。



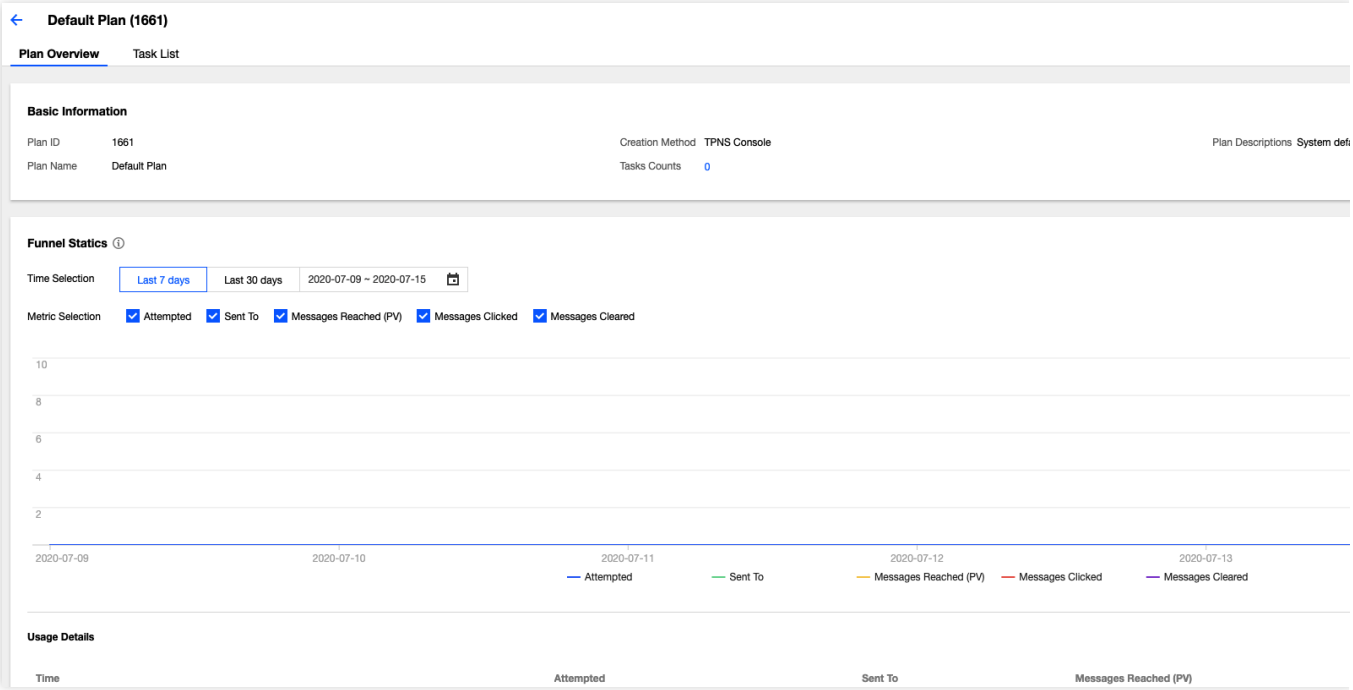

## 在推送计划中创建推送任务

在推送计划详情中，选择**任务列表**页签，单击**新建推送**。




## 查看推送计划的聚合数据统计

在推送计划详情中，选择**计划概览**页签，即可查看推送计划的汇总数据统计。

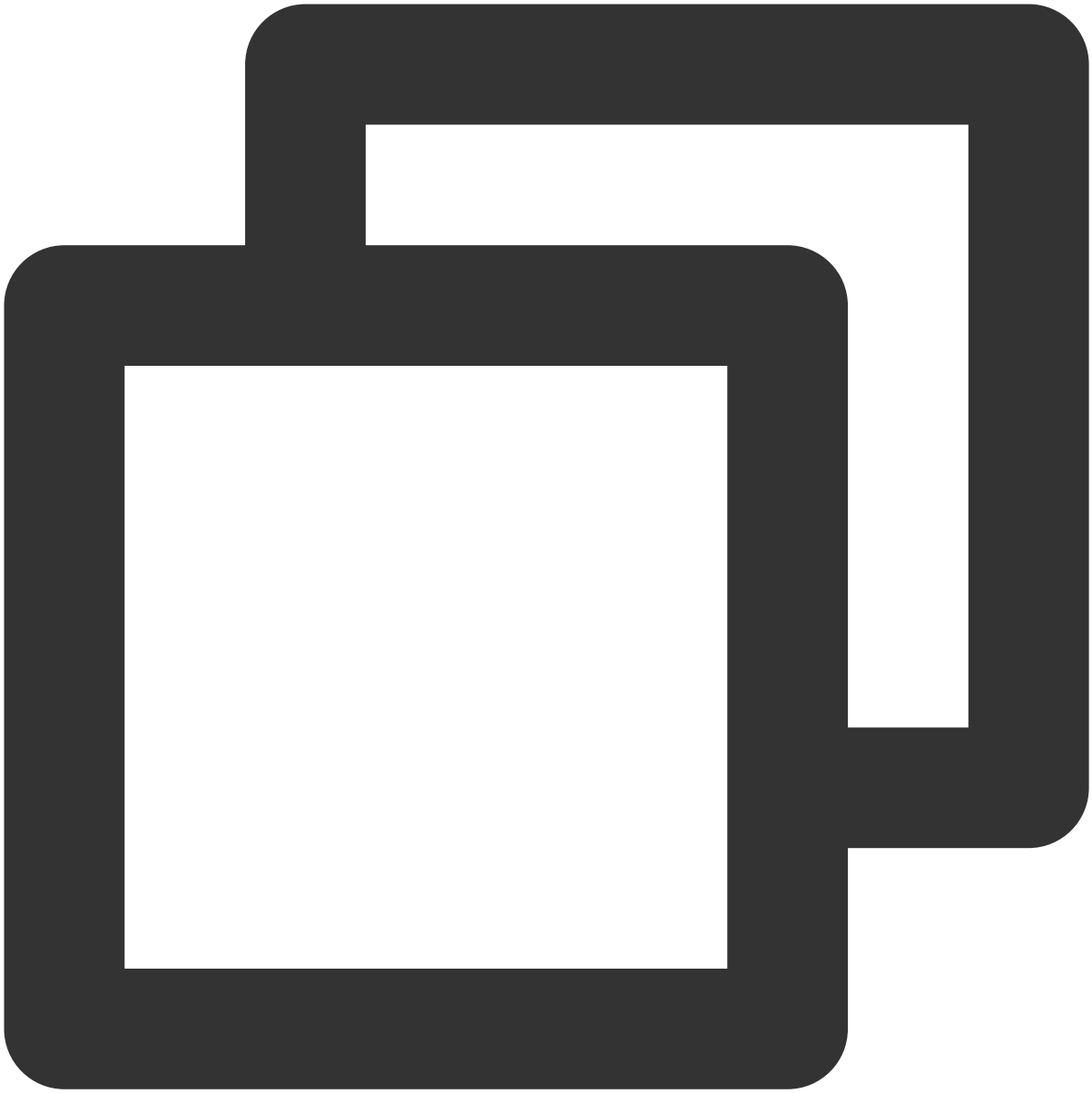



## REST API 使用（创建推送计划、推送时指定推送计划）

### 创建推送计划

参考 创建推送计划接口 创建您的推送计划，创建完成后，可指定该推送计划发送消息。

### 创建示例

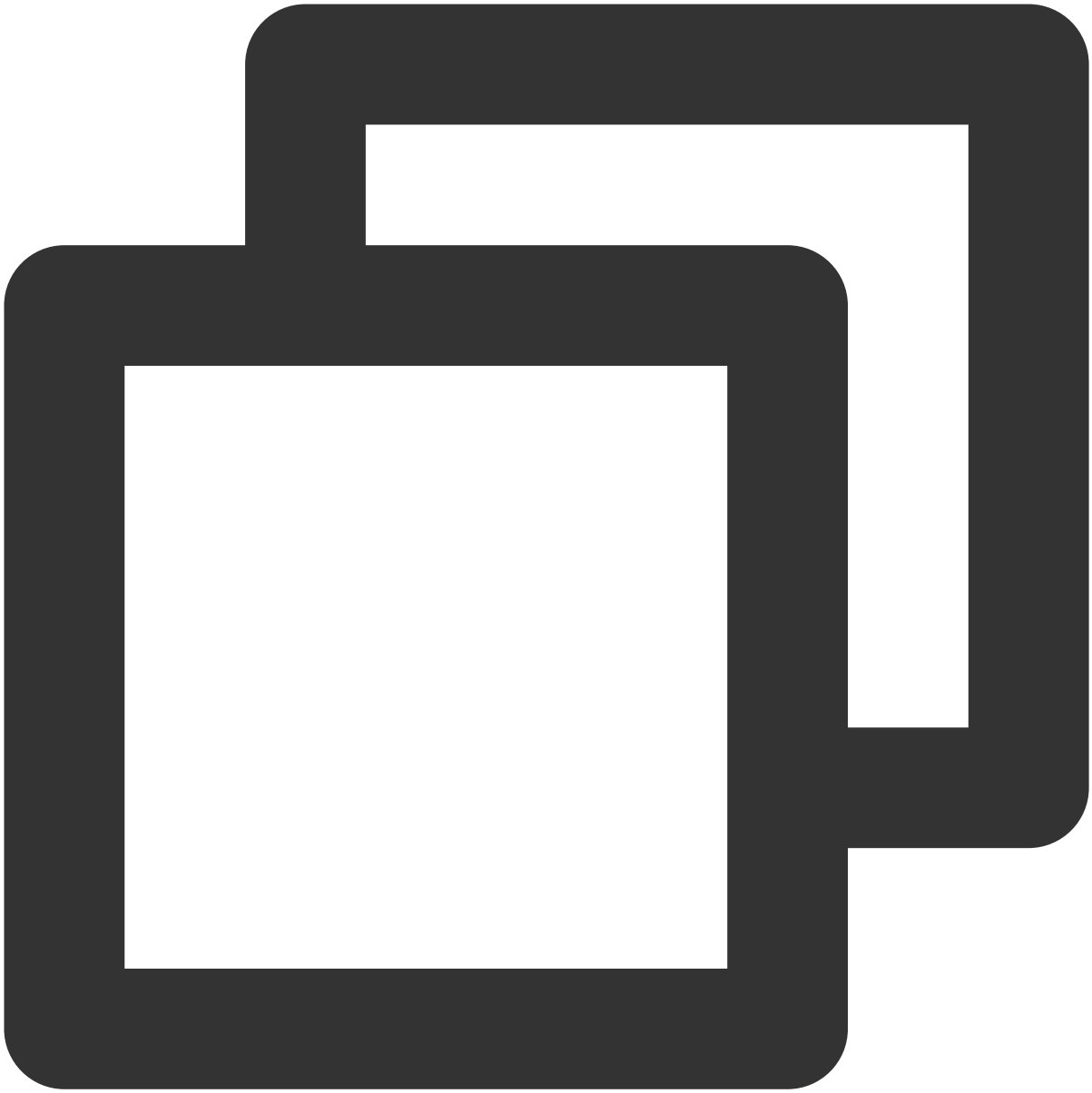


```
{
            "planName":"VIP_Level 15",
            "planDescribe":"VIP会员福利派送"
}
```

**指定推送计划推送**

在调用推送接口时，可以通过设置 `plan_id` 来指定推送计划 ID 来发送给目标用户群体，详情可参考 REST API 可选参数说明。

推送示例如下：



```
{
    "audience_type": "token",
    "environment":"dev",
    "token_list": [ "05da87c0ae********fa9e08d884aada5bb2"],
    "message_type":"notify",
    "plan_id":"20200704",
    "message":{
     "title": "推送标题",
    "content": "推送内容",
    "ios":{
        "aps": {
```

```
            "alert": {
                "subtitle": "推送副标题"
            },
            "badge_type": -2,
            "sound":"Tassel.wav",
            "category": "INVITE_CATEGORY"
        },
       "custom_content":"{\\"key\\":\\"value\\"}",
    }
 }
}
```

# 推送控制

最近更新时间：2024-01-16 17:34:39

## 功能介绍

消息推送是一把双刃剑。如果合理运用，可以以极低的成本快速获取大量的用户活跃、运营转化等收益；反之，若使用不合理，则会对产品口碑产生负面影响，甚至会导致卸载量突增。
基于以上背景，移动推送提供以下能力保障您的推送安全和推送效率：
推送限频——防止过度推送引起用户反感。
防止重复推送——从系统层面防止因人工操作失误导致的重复推送。
可接收推送时段——可设置用户可接收推送的时段，防止深夜滋扰用户。

## 使用说明

### 推送限频

#### 操作背景

电商类 App 经常需要产品推广和营销活动，App 通过浏览习惯、购物频率等信息圈选目标人群，进行定向消息推送，同一个用户可能在短时间内收到多条推送，对用户体验造成严重影响，甚至导致用户关闭通知栏或直接卸载应用。

#### 操作步骤

1. 登录 移动推送控制台。
2. 在左侧导航栏选择 **App推送管理** > **推送控制**。
3. 在**推送限频**一栏中，单击**开启**。
4. 设置单设备单日接收推送条数上限，可设置1-100内整数，默认为3条，并选择参与限频的 推送计划，单击**保存**，完成配置。

### 防止重复推送

#### 操作背景

资讯、视频类 App 通常需要对突发新闻热点进行抢先报道，并且完成用户的快速触达，从而帮助用户快速获得最新、关注度最高的资讯内容，若出现短时间内重复推送，会极大损伤用户体验，开启此功能可避免用户在短时间内收到重复内容的推送。

#### 操作步骤

1. 登录 移动推送控制台。
2. 在左侧导航栏选择 **App推送管理** > **推送控制**。
3. 在**防止重复推送**一栏中，单击**开启**。

开启后，系统自动拦截一小时内「推送目标与推送文案」完全相同的推送，防止用户在短时间内收到重复推送。

**说明：**

仅限制推送目标为「所有设备」、「标签组合」、「批量账号」、「用户分群」的推送。

## 可接收推送时段

### 操作背景

工具类或具有系统通知类需求的 App，通常在用户或系统功能状态变化时，需要对用户进行通知，但在深夜时段推送会滋扰 App 用户，进而影响 App 的口碑，甚至提高卸载率，可配置用户接收推送时段，从而减少对用户的打扰。

### 操作步骤

1. 登录 移动推送控制台。
2. 在左侧导航栏选择 **App推送管理** > **推送控制**。
3. 在**可接收推送时段**一栏中，单击**开启**，设置终端用户可以接收推送的时间段，单击**保存**完成配置。在可接收时段外的推送，将不再下发。

**说明：**

**可接收推送时段**控制的是推送可下发的时段，而不是终端设备可接收的时段。如可接收推送时段设置为「15:00-16:00」，那在这期间创建的推送任务都可以下发；这个时间段之外创建的推送任务，移动推送会进行拦截。

# 消息回撤

最近更新时间：2024-01-16 17:34:39

## 使用场景

若 App 已发送的通知内容或跳转链接有误，终端用户查看或点击后会对产品有负面影响，此时需要及时处理。您可以选择终止、撤回该消息或用新消息覆盖旧消息，如有定时任务的，您也可以及时取消推送继续下发。本文通过 控制台 和 Rest API 两种方式为您介绍如何进行消息终止、撤回、覆盖以及取消定时推送。

## 功能描述

终止：终止该任务离线保存有效期内的所有离线消息。
撤回：终止该消息并使已发送但未被点击或清除的通知在终端设备通知中心消失，实现无感撤回的效果。
覆盖：终止该消息并用一条新通知覆盖掉当前通知。覆盖成功后，终端设备通知中心只展示新通知。
取消定时任务：对于尚未调度的定时任务，进行取消操作。取消成功后，定时推送任务不会再下发。

## 使用限制

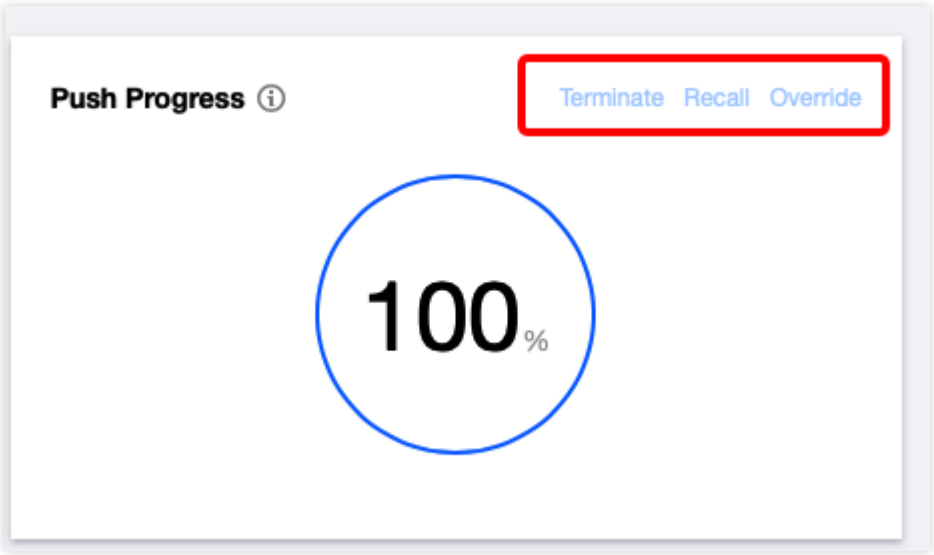
| 平台 | 终止 | 撤回 | 覆盖 | 取消定时任务 |
| --- | --- | --- | --- | --- |
| Android | 支持 | 该功能还在升级中，暂未上线 | 仅支持移动推送自建通道、小米通道、魅族通道以及华为系统版本 EMUI10 及以上的设备 | 支持 |
| iOS | 不支持 | 该功能还在升级中，暂未上线 | 支持 | 支持 |

**说明：**
消息终止/撤回/覆盖仅支持推送目标为**所有设备**、**批量账号**与**标签**的通知。
取消定时任务仅支持**所有设备**、**号码包**与**标签**的通知

## 控制台使用

**取消终止、覆盖或撤回操作**

1. 登录 移动推送控制台。
2. 在左侧的菜单栏中，选择 **App 推送管理** > **推送任务**，进入推送任务页面。
3. 在需要终止/覆盖/撤回的推送任务处，单击**查看详情**。
4. 在推送详情页面右上角**推送进展**一栏中，选择对该任务进行终止、覆盖或撤回操作。




**说明：**

当您选择消息覆盖时，对于不支持消息覆盖的 vivo、OPPO 以及华为系统版本 EMUI10 以下的设备，您可选择是否继续进行消息下发：

若选择继续下发，则会覆盖通知，新消息和原消息会同时出现在设备的通知中心。

若选择不下发，则对上述不支持覆盖的设备不下发新通知。

5. 操作成功后，可返回**推送任务**页面，在状态一栏查看该任务当前状态。

### 取消定时任务操作

1. 登录 移动推送控制台。
2. 在左侧的菜单栏中，选择 **App 推送管理** > **推送任务**，进入推送任务页面。
3. 在需要取消定时任务的推送任务后，单击**取消推送**即可取消。
4. 操作成功后，状态一览可查看该推送任务为 取消 状态。

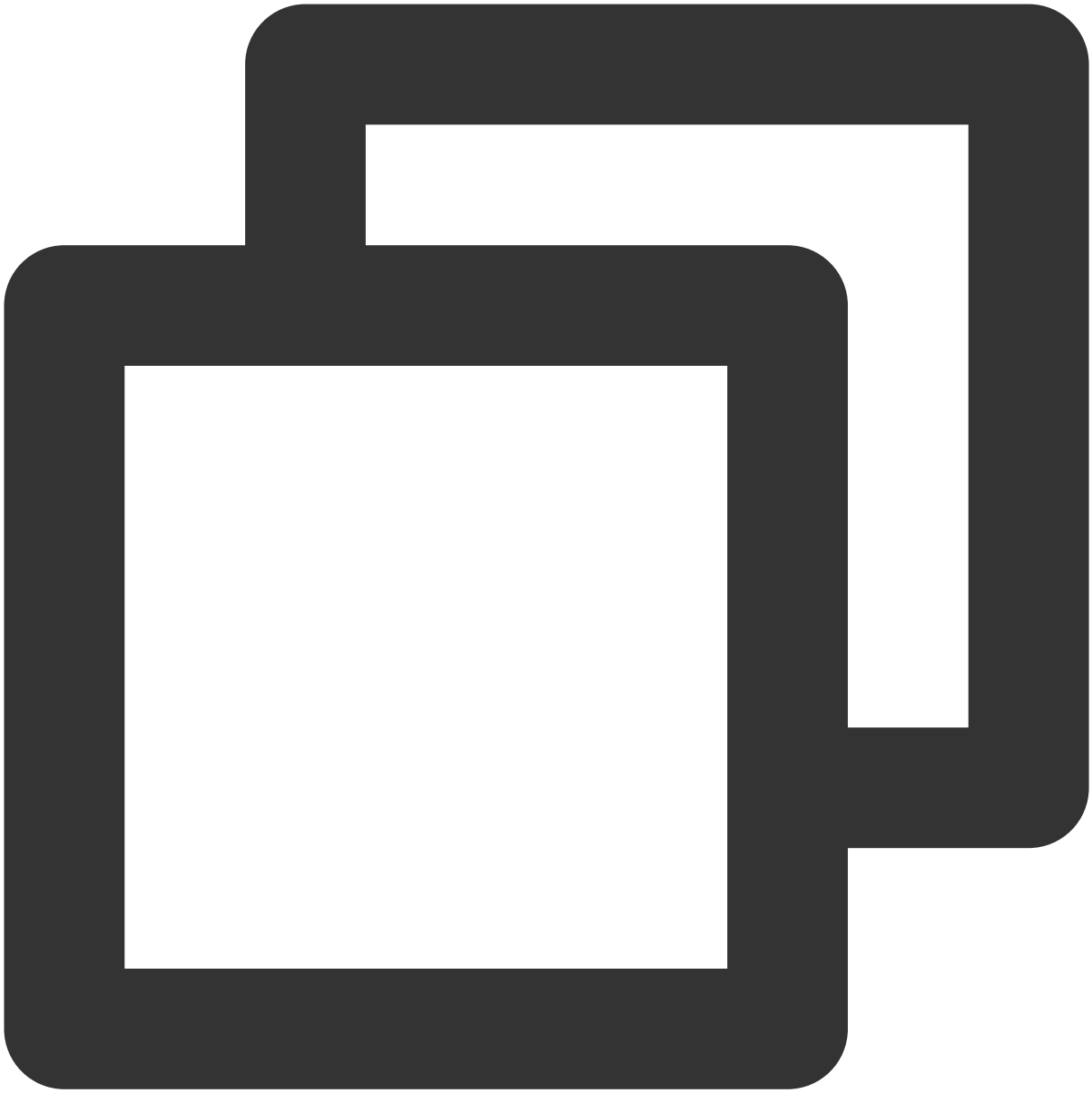
## Rest API 使用

### 消息终止

#### 接口说明

请求方式：POST

请求地址：



```
服务地址/v3/push/stop_push_msg
```

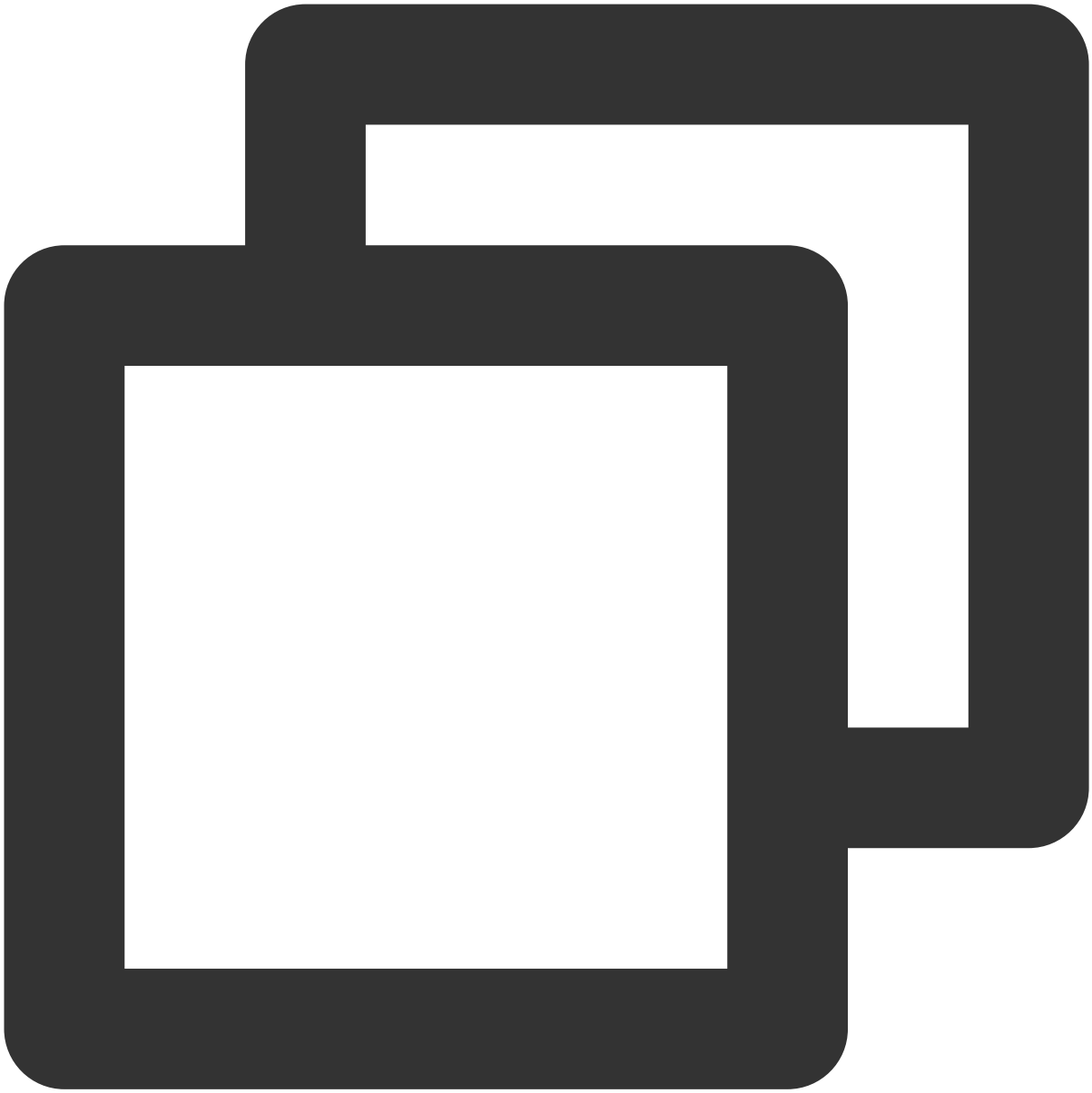
接口服务地址与服务接入点一一对应，请选择与您的应用服务接入点对应的 服务地址。

**请求参数**

| 参数名 | 类型 | 是否必须 | 参数说明 |
| --- | --- | --- | --- |
| pushId | string | 是 | 推送任务的任务 ID |

版权所有：腾讯云计算（北京）有限责任公司

**请求示例**



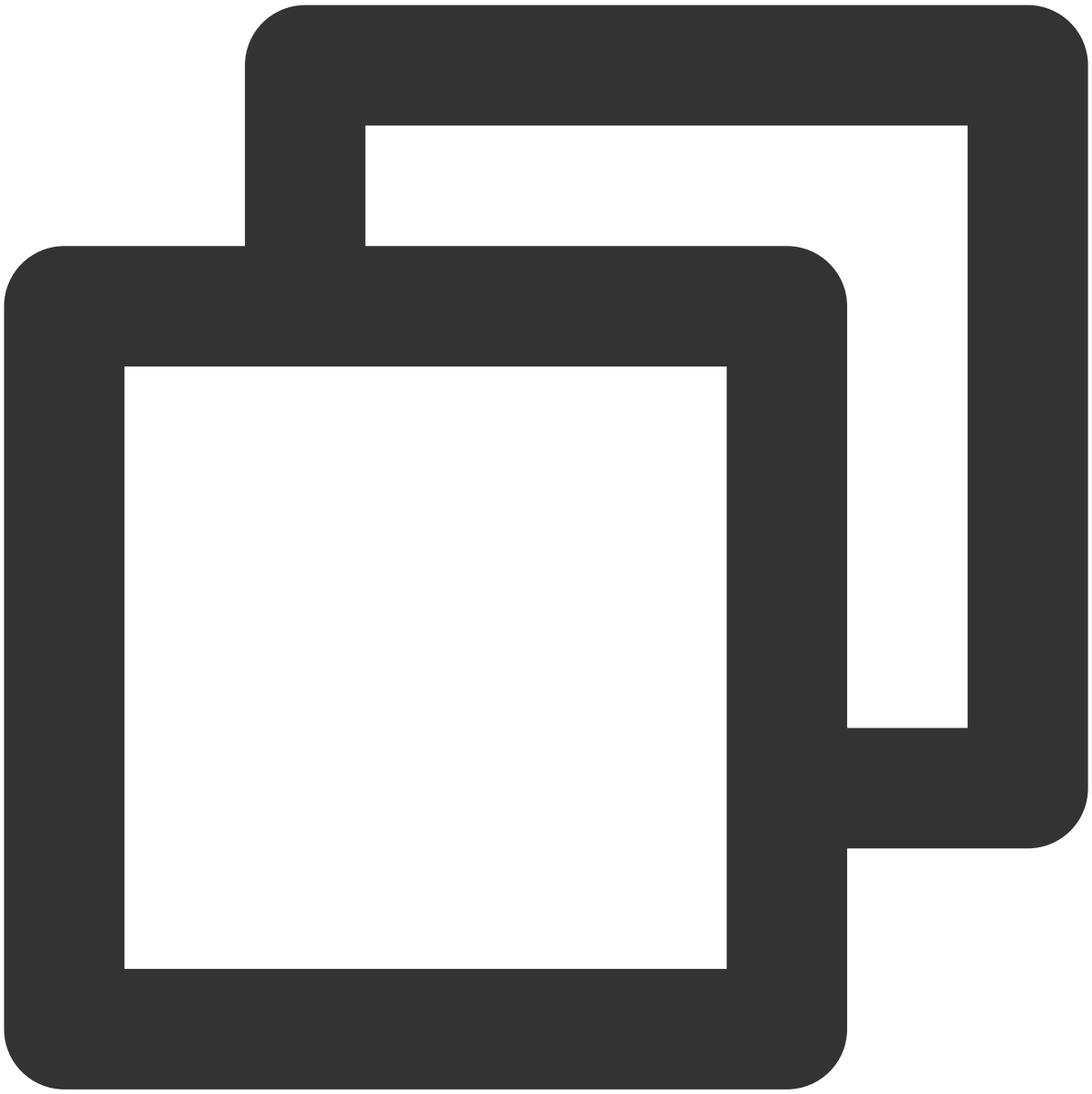
```
{
            "pushid":"43214535"
}
```

## 消息撤回（该功能还在升级中，暂未上线）

**接口说明**

请求方式：POST

请求地址：



```
服务地址/v3/push/revoke_push_msg
```

接口服务地址与服务接入点一一对应，请选择与您的应用服务接入点对应的 服务地址。

**说明：**

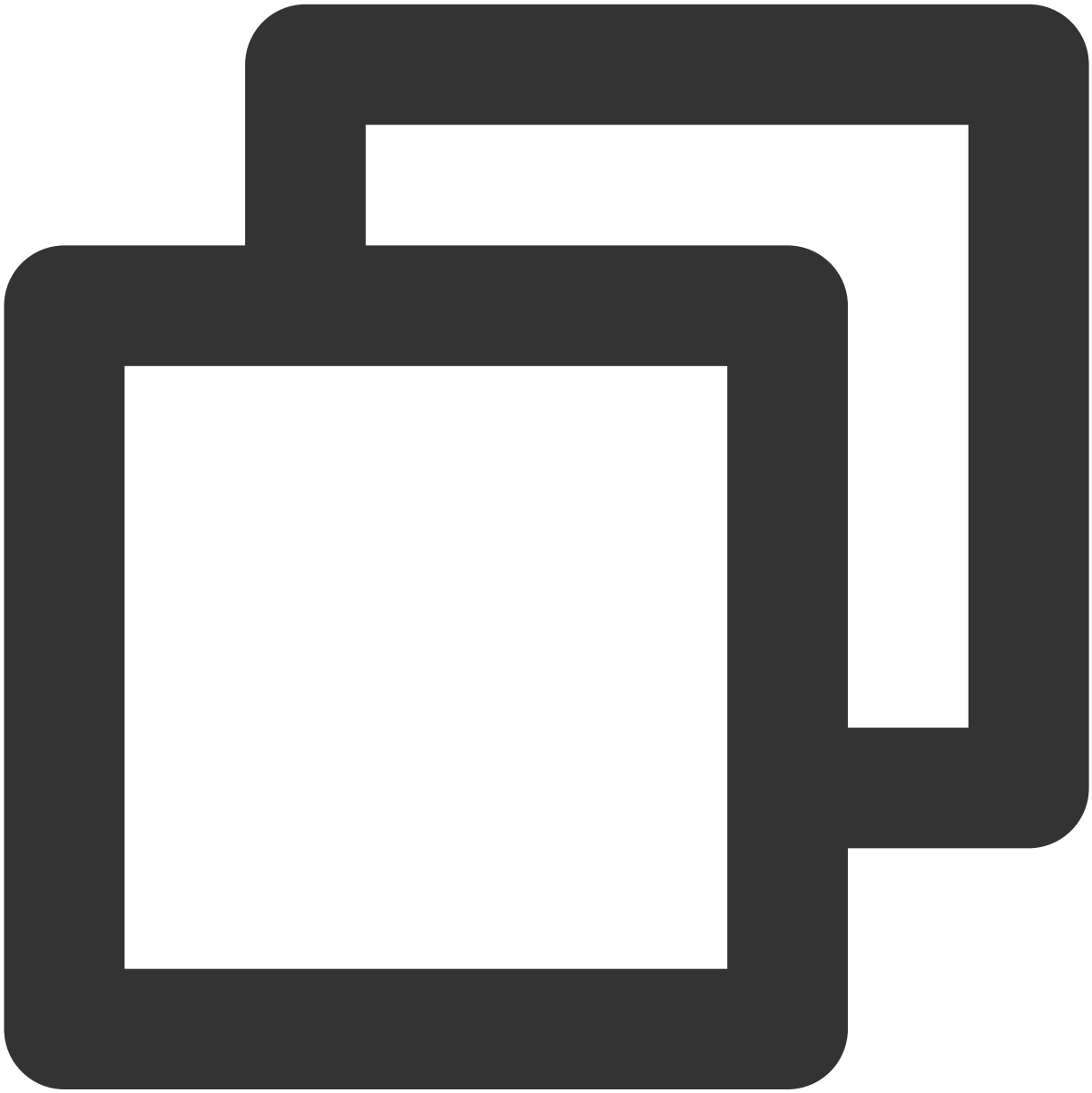
撤回消息会默认终止该推送任务的离线消息下发。

**请求参数**

| | | | |
|---|---|---|---|

| 参数名 | 类型 | 是否必须 | 参数说明 |
| --- | --- | --- | --- |
| pushId | string | 是 | 推送任务的任务 ID |

**请求示例**



```
{
            "pushid":"150032"
}
```

## 消息覆盖

### 步骤1：查询推送任务的 `collapse_id`

调用 单个任务推送信息查询接口，在应答参数中获取 `collapse_id`，例如0001。

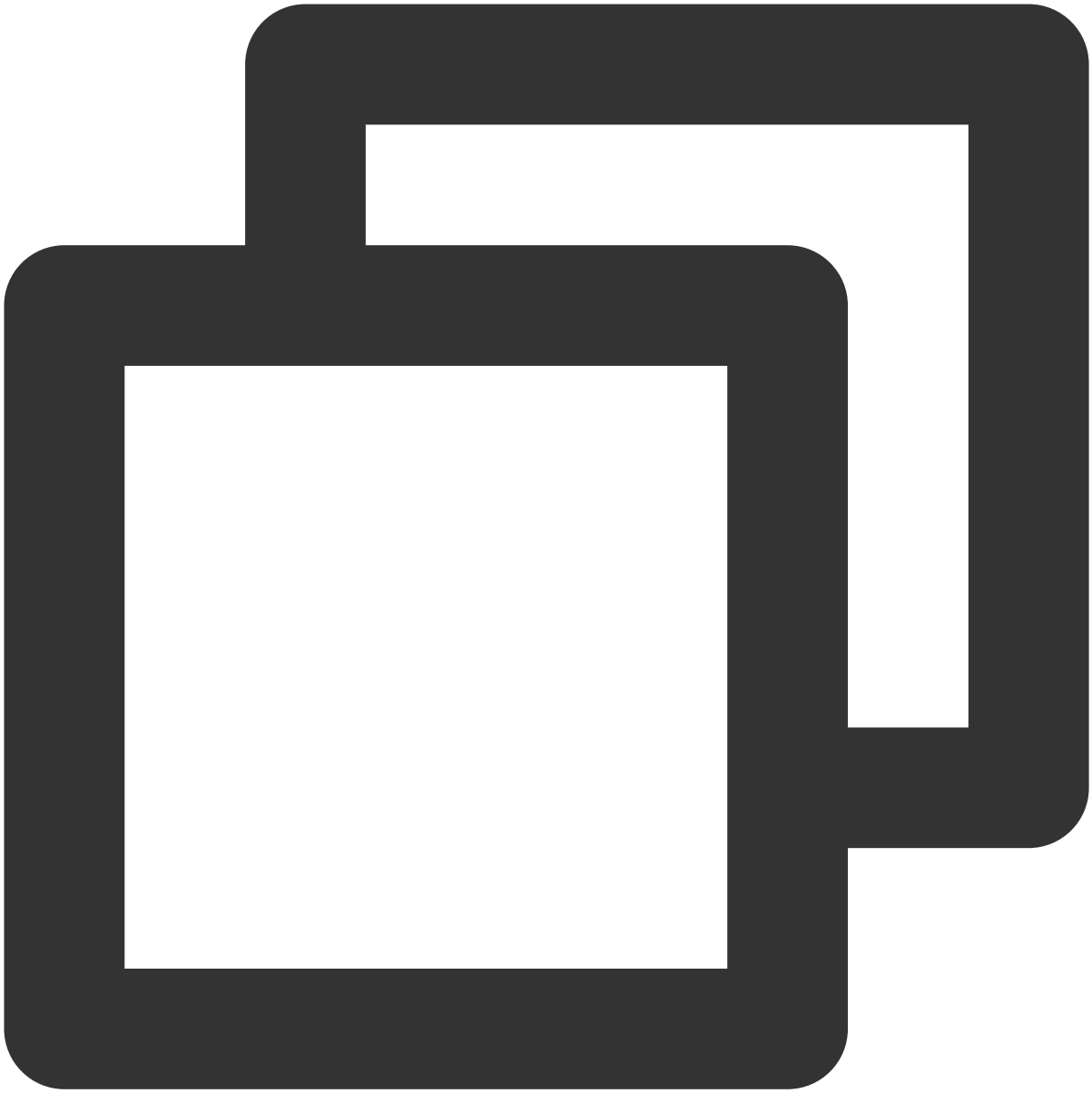
### 步骤2：调用推送接口覆盖原推送内容

在您调用 推送接口 时， 补充上述**步骤1**中获取的 `collapse_id`，例如0001，并且可通过设置 `force_collapse` 字段来决定对于不支持消息覆盖的设备，是否进行消息下发。

**说明：**

覆盖消息会默认终止原推送任务的离线消息下发。

### 推送示例



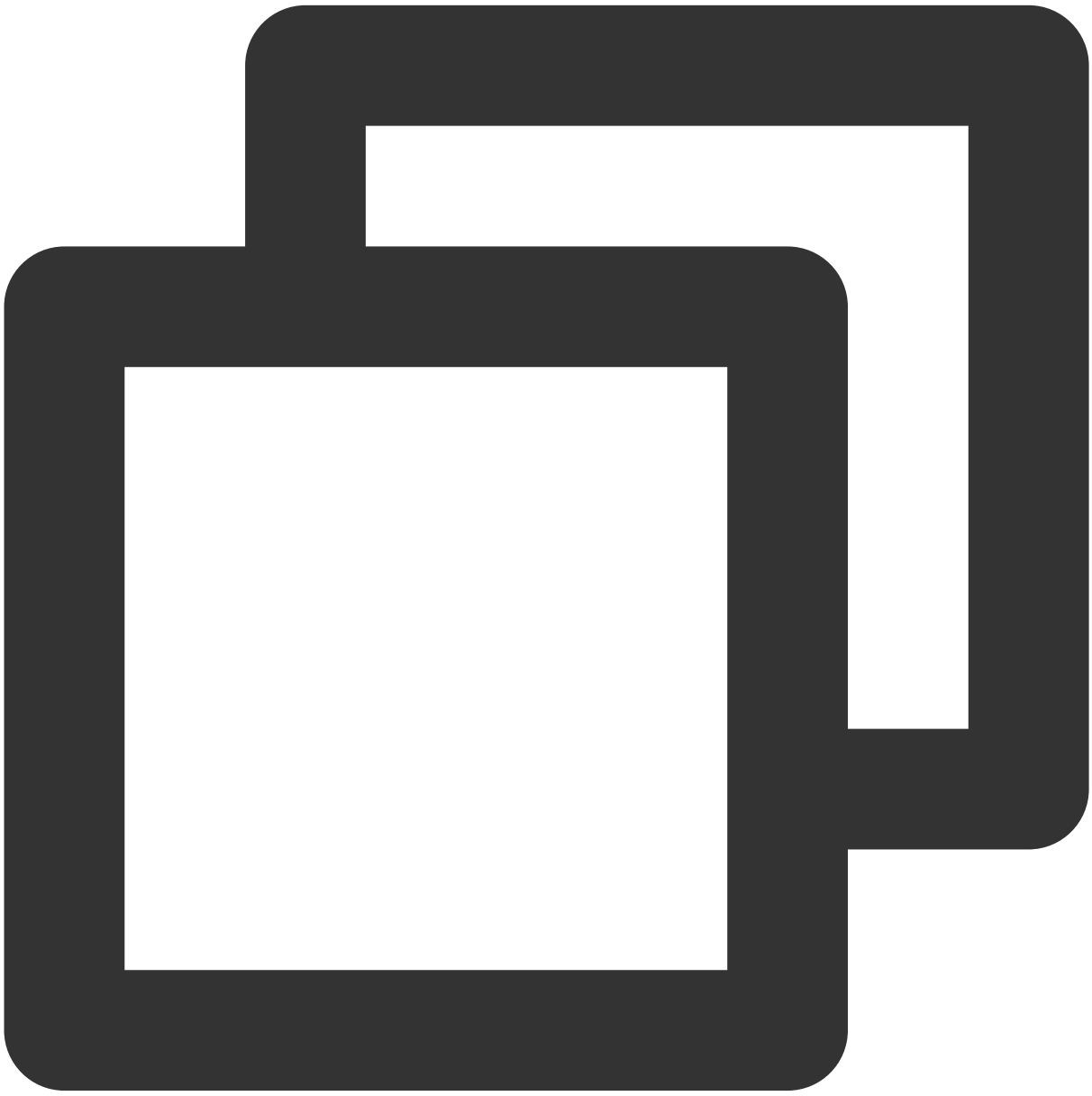
```
{
    "audience_type": "all",
    "collapse_id": 0001,
      "force_collapse":false,
    "message_type": "notify",
    "message": {
        "title": "覆盖00001条消息",
        "content":"今天天气不错哦"
    },
    "platform": "android"
}
```

## 取消定时任务

### 接口说明

请求方式：POST

请求地址：



```
服务地址/v3/push/cancel_timing_task
```

接口服务地址与服务接入点一一对应，请选择与您的应用服务接入点对应的 服务地址。

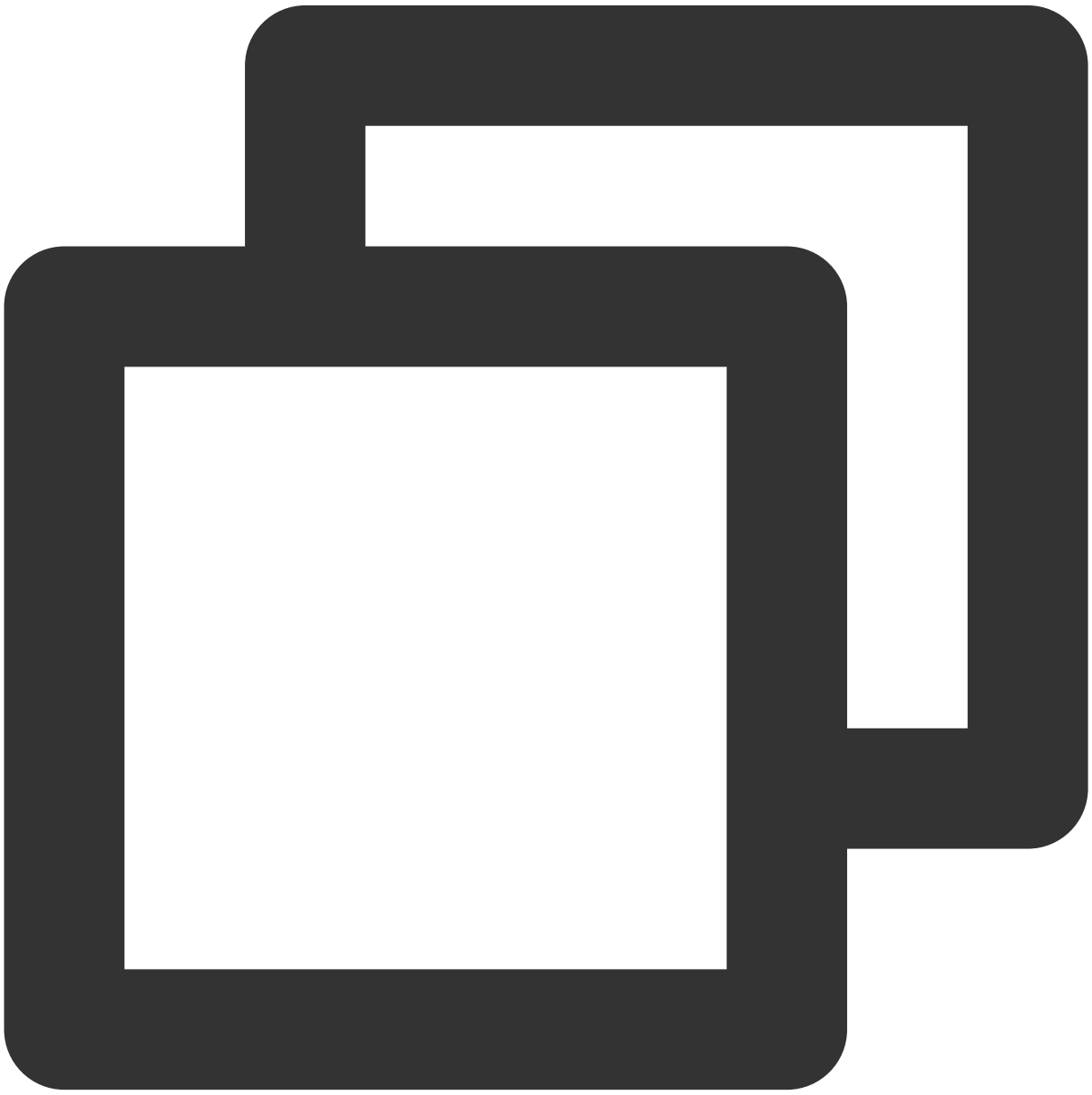
**说明：**

对于尚未调度的定时任务，进行取消操作。

**请求参数**

| 参数名 | 类型 | 是否必须 | 参数说明 |
| --- | --- | --- | --- |
| pushId | string | 是 | 推送任务的任务 ID |

**请求示例**



```
{
            "pushid":"15003211"
}
```

版权所有：腾讯云计算（北京）有限责任公司

# 推送高级功能
# 标签功能

最近更新时间：2024-01-16 17:34:39

标签是精准推送中会用到的功能，开发者可以调用移动推送SDK 或服务端 API，为设备绑定一个/多个标签。在完成绑定之后，就可以针对标签来进行推送，方便进行更精细化的运营。

**说明：**

标签推送属于群推，会有限频，不建议频繁推送或标签对应设备较少的情况下使用标签推送，可以相应的使用其他单推或列表推。

## 标签推送场景

### 用户促活与流失召回

App 运营中经常需要针对新用户进行消息提醒，从而作为新用户体验旅程设计的重要一环，提高新用户的留存率；通过在推送时选择移动推送提供的“设备新增”标签，即可快速对指定日期用户进行推送，方便快捷。

同样，我们也提供了“沉默用户”标签，可以指定 N 日内未活跃的用户作为推送对象，对其进行消息推送将这些流失用户进行召回，提升 App 活跃用户数。

### 赛事订阅通知

某直播 App 10月24日18:00有一个足球比赛直播，10月20日开始开放直播预约，希望在直播开始前，给订阅这个节目的用户推送直播即将开始的通知。

在用户订阅这个节目时，可以将这个节目的主题"10241800足球"作为标签和这个用户设备 token 进行绑定，当直播快开始时，选择"足球"这个标签进行比赛即将开始的推送，当比赛完毕后，再调用移动推送的解绑标签接口，将"10241800足球"标签和设备 token 进行解绑。

### 续费通知

应用 A 希望对3天后会员到期的用户推送续费通知，假设当前某设备 token 绑定的标签为"football"和"deadline:20200210"，用户在2020年2月9号续期了一个月，则此时需用"deadline:20200310"的标签将"deadline:20200210"替换掉，即 deadline 这个标签类别只能有一个值（最新值），此时可调用移动推送提供的 kv 覆盖接口，在不影响其他标签的情况下，解绑"deadline:20200210"标签之后，再绑定"deadline:20200310"标签，推送时（可设置定时推送），选择标签为当天时间+3天的标签，如当天日期是20200307，则可选择标签为"deadline:20200310"进行续费通知的推送。

## 标签概览

版权所有：腾讯云计算（北京）有限责任公司

移动推送目前提供了两大类标签功能，包括用户自定义标签和移动推送平台侧维护的预设标签，标签类别如下表所示：

| 标签类型 | 场景 | 移动推送内置 tag_type | 限制 | 示例 |
| --- | --- | --- | --- | --- |
| 自定义标签 | 客户自定义，例如会议 ID，班级 ID，用户喜好（篮球，数码产品等）等 | xg_user_define | 最多可以有10000个自定义 tag（如需提高该限制，请联系 在线客服）<br>每个设备 token 最多可绑定100个自定义 tag（如需提高该限制，请联系 在线客服)<br>每个自定义 tag 可绑定的设备 token 数量无限制 | love_basketball, love_shopping, male 等 |
| 系统预设标签 | App 版本 | xg_auto_version | 移动推送系统预设，无限制 | 1.0.1，1.0.2等 |
| | 省份 | xg_auto_province | | guangdong, hunan，shanghai等 |
| | 城市 | xg_auto_city | | beijing，tianjin, chongqing等 |
| | 活跃信息 | xg_auto_active | | 20200521, 20200522等 |
| | 移动推送 | xg_auto_sdkversion | | |

| | | | |
|---|---|---|---|
| SDK 版本 | | | 1.1.5.4，1.1.5.4，1.1.6.1等 |
| 系统版本 | xg_auto_systemversion | | 10.0.0, 12.4.5等 |
| 系统语言 | xg_auto_systemlanguage | | zh，en，ja 等 |
| 国家 | xg_auto_country | | CN，US 等（大写字母） |
| 手机品牌 | xg_auto_devicebrand | | xiaomi，huawei 等 |
| 机型 | xg_auto_deviceversion | Samsung Note4, vivo Y75A 等 | - |
| 最近连续活跃 | 暂不支持 API 调用 | 最近 N 天活跃的设备，N 取值为[1,30]的任意一值，格式为 string。 | 最近"10"天活跃的设备 |
| 最近连续沉默 | 暂不支持 API 调用 | 最近 N 天不活跃的设备，N 取值为[1,30]的 | 最近"10"天不活跃的设备 |

| | | | | |
|---|---|---|---|---|
| | | | 任意一值，格式为 string。 | |
| 最近注册 | 暂不支持 API 调用 | 最近注册的设备，标签取值为起始日期至截止日期 [startDate, endDate]，格式 [YYYYmmdd,YYYYmmdd]<br>startDate 和 endDate 之间的日期间隔不能超过30天<br>endDate 不能为当天<br>startDate 不能超过90天<br>startDate 和 endDate 可以相同 | [20200901,20200910] 注册的设备 | - |

**说明：**

在使用 API 进行标签推送时需使用移动推送内置 tag_type 设置标签类型。

# 使用前准备

## 自定义标签管理

自定义标签是指用户可以自己定义设备标签名字的标签设置方式，目前移动推送提供了两种方式，一种是用 REST API 进行标签设置，另一种是通过终端 SDK 进行标签设置。

**方式一：通过 rest API 进行标签设置**

自定义标签的绑定和解绑。

API 具体参见文档：标签绑定

**方式二：通过终端 SDK 进行标签设置**

iOS SDK 进行标签的设置，详情请参见文档 自定义标签设置。

Android SDK 进行标签的设置，详情请参见文档 自定义标签设置。

**说明：**

单个设备最多可绑定100个标签（超出可联系 在线客服 申请提高额度）。

单个 App 最多可绑定10000个标签（超出可联系 在线客服 申请提高额度）。

单个标签不能超过50个字节。

批量绑定解绑标签时，单次请求标签数量不能超过500个。

### 自定义标签使用场景及关键字

标签推送适用于标签绑定设备多（一般绑定设备数超过10个），且推送频次较少（一般该标签的日推送数不超过10次）的场景，针对绑定设备数较小且推送频次较多的标签推送场景，推荐使用账号推送，即通过账号绑定多个设备，用账号替代标签来进行推送。

**关键字**

冒号":"是标签绑定时的关键字，用户标签的 key-value 分割，例如用户给某个设备 token 打了"level:3"的标签，则移动推送后台会将"level"作为这个标签的 key，"3"作为标签的 value，这个不影响用户的实际推送，还是按"level:3"这个标签来推，key-value 方式存储主要是方便后续的同类标签的覆盖，如这个设备 token 还绑定了其他标签，如"male"，即此时该设备对应两个标签，"male"和"level:3"，如果这个设备对应的会员等级升级到4级了，用户希望用"level:4"标签来覆盖"level:3"标签，而不影响"male"标签，此时可直接调用 kv 覆盖接口来给这个设备设置"level:4"标签，这个设备 token 所绑定的标签则变为"male"和"level:4"。

**绑定/解绑标签**

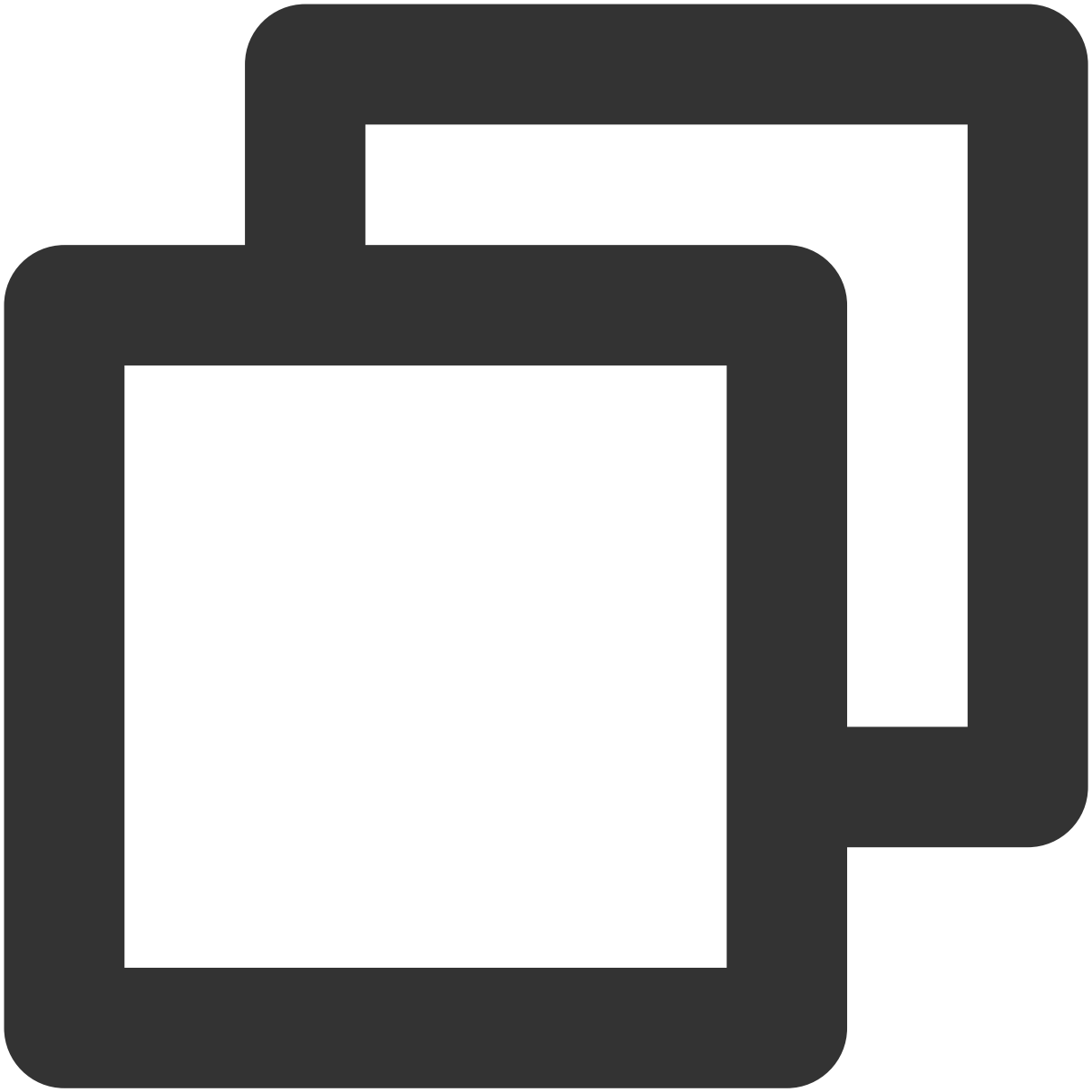
移动推送提供了单标签绑定/解绑单设备，单标签绑定/解绑多设备，多标签绑定/解绑单设备，多标签绑定/解绑多设备的接口。

**单标签单设备**

**推荐场景**

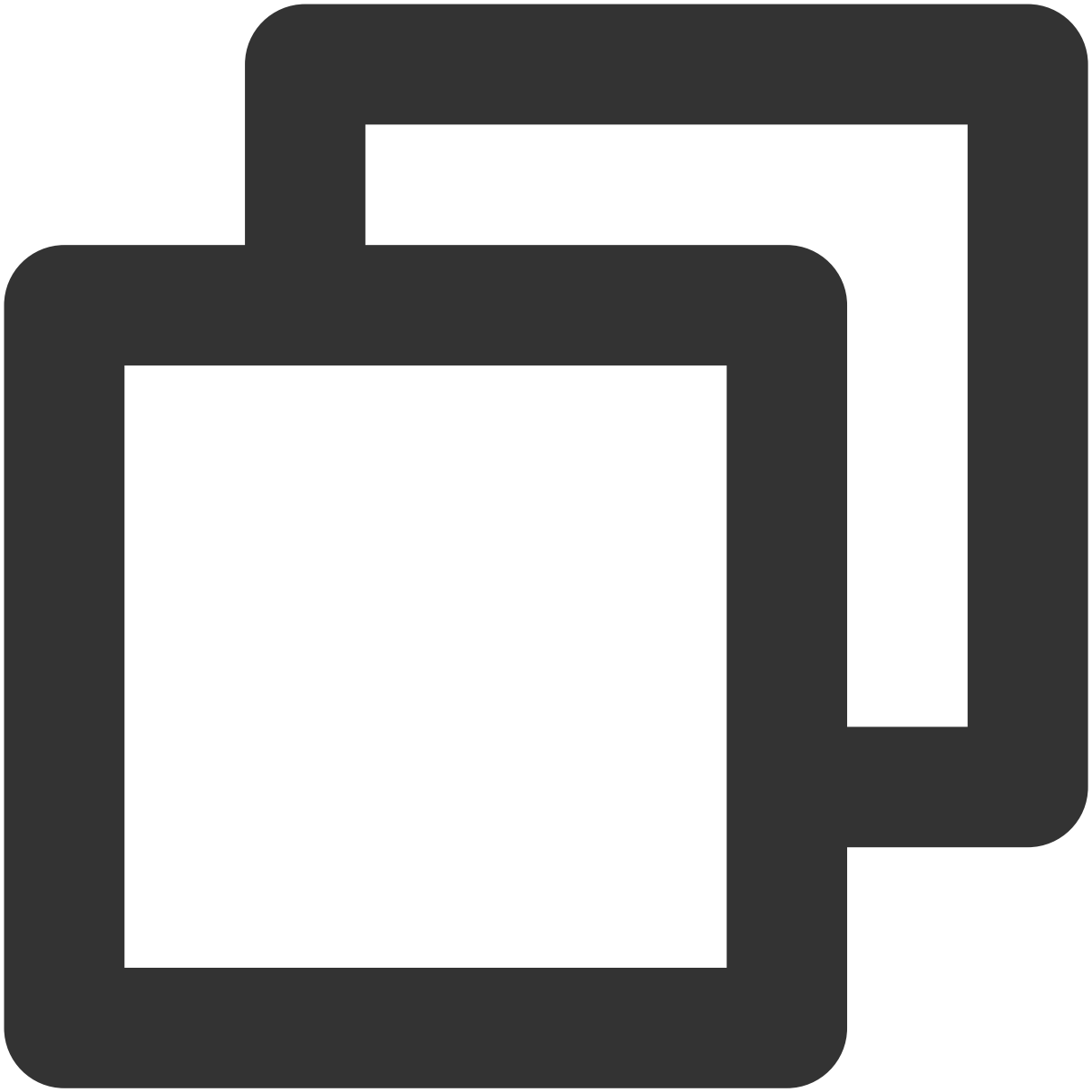
1. 调用终端 SDK 接口，如 App 内自动获取用户订阅频道，对该频道标签和设备 token 进行绑定，反之进行解绑。
2. 非频繁的调用 Rest API 接口，如接入测试时的调用。

**标签绑定用法**



```
{
    "operator_type": 1,
    "tag_list": ["tag"],
    "token_list": ["token"]
}
```

**标签解绑用法**



```
{
    "operator_type": 2,
    "tag_list": ["tag"],
    "token_list": ["token"]
}
```

**使用限制**

每个标签最大50个字节。

接口调用逻辑为同步。

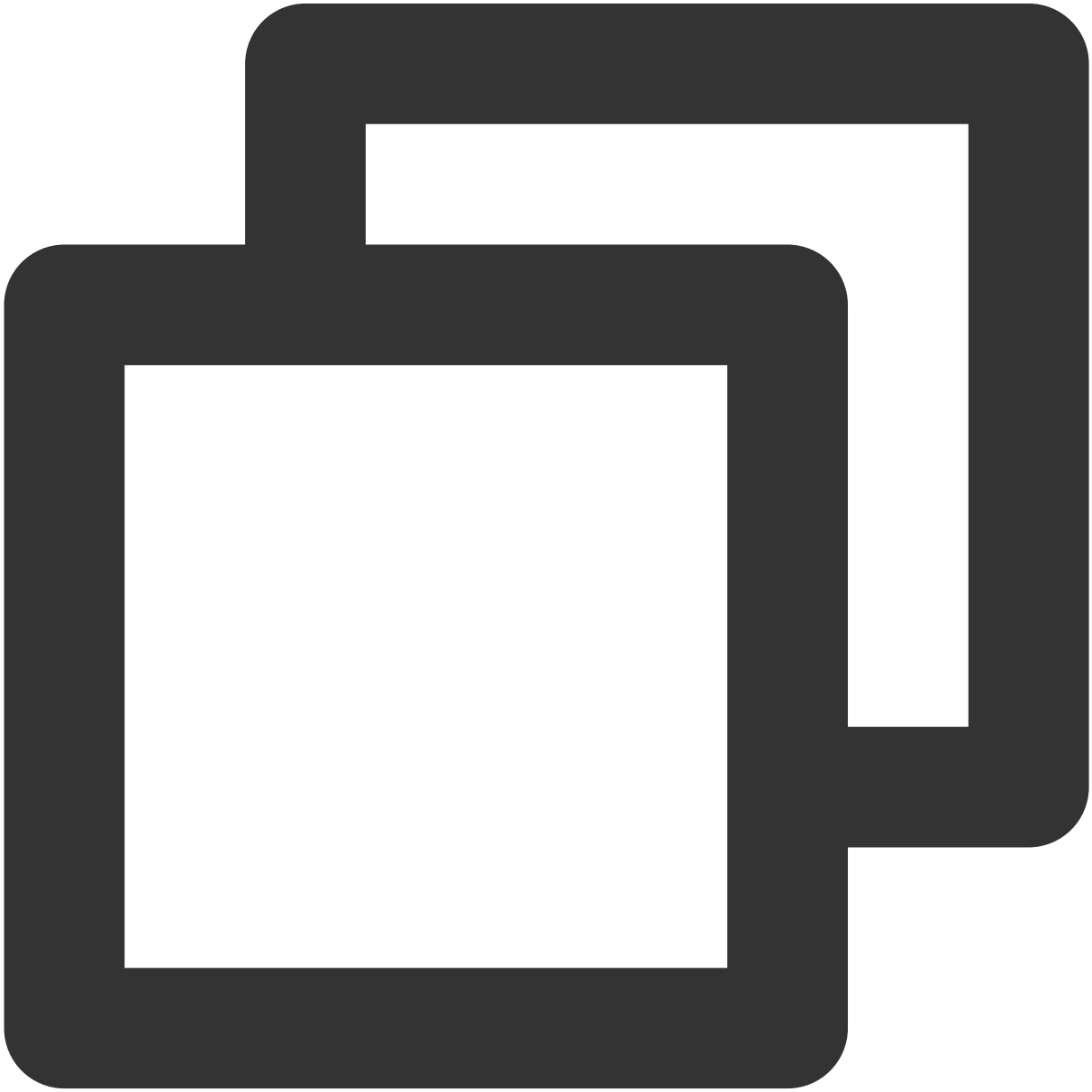
**多标签单设备**

**推荐场景**

1. 调用终端 SDK 接口，如 App 内自动获取用户特征信息，如年龄，省份，性别等标签信息，对这些标签和用户设备 token 进行绑定，反之进行批量解绑。

2. 调用 Rest API 接口，如 App 运营人员从公司内其他渠道获取该设备对应的订阅信息，如单身，爱好足球，喜欢看电影等，对这些标签和该用户设备 token 进行绑定，反之进行批量解绑。

**标签绑定用法**

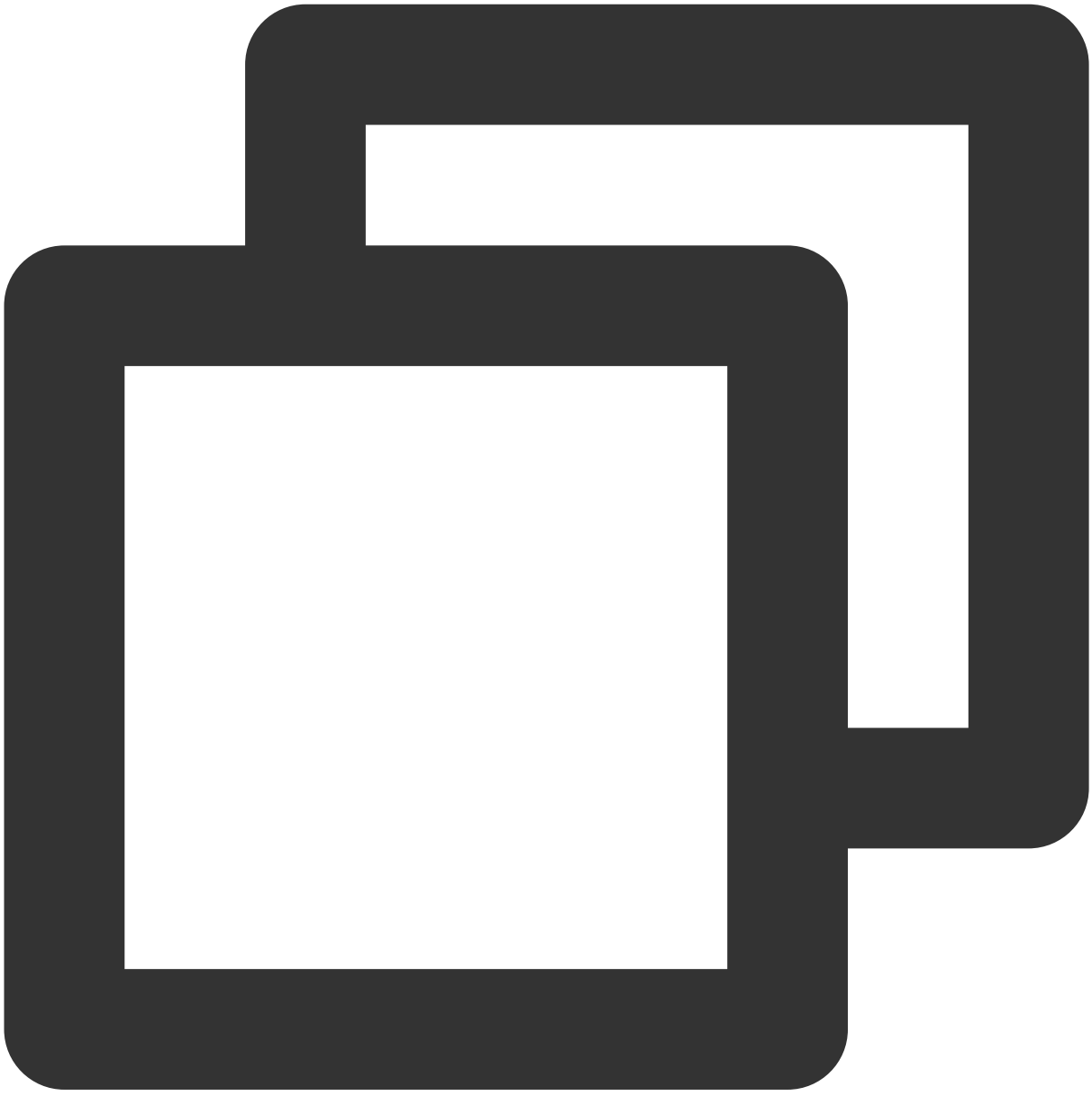


```
{
    "operator_type": 3,
```

版权所有：腾讯云计算（北京）有限责任公司

```
    "tag_list": ["tag1","tag2"],
    "token_list": ["token"]
}
```

**标签解绑用法**



```
{
    "operator_type": 4,
    "tag_list": ["tag1","tag2"],
    "token_list": ["token"]
}
```

**使用限制**

每个标签最大50个字节。

每次调用标签数量最大100个。

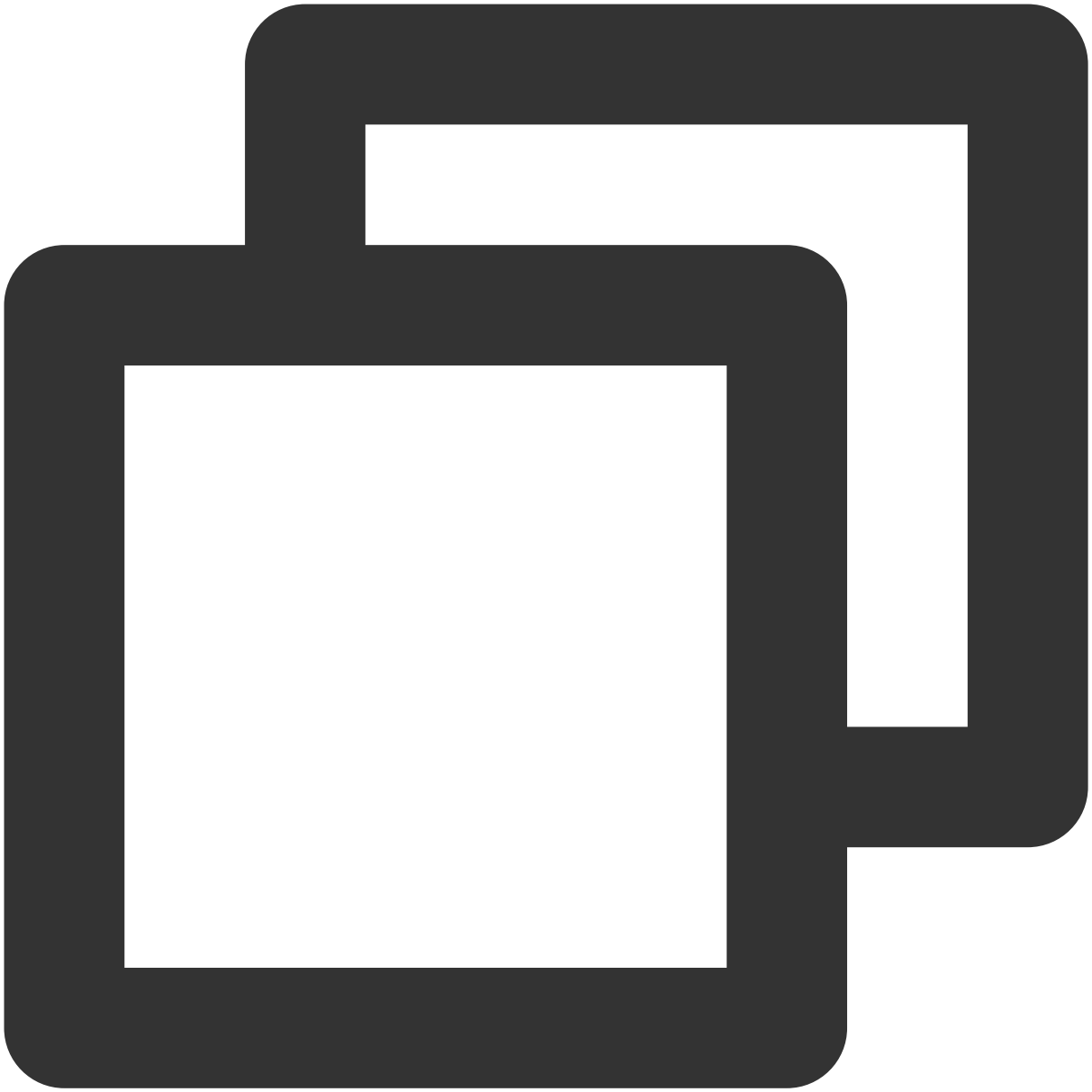
接口调用逻辑为同步。

#### 单标签多设备

**推荐场景**

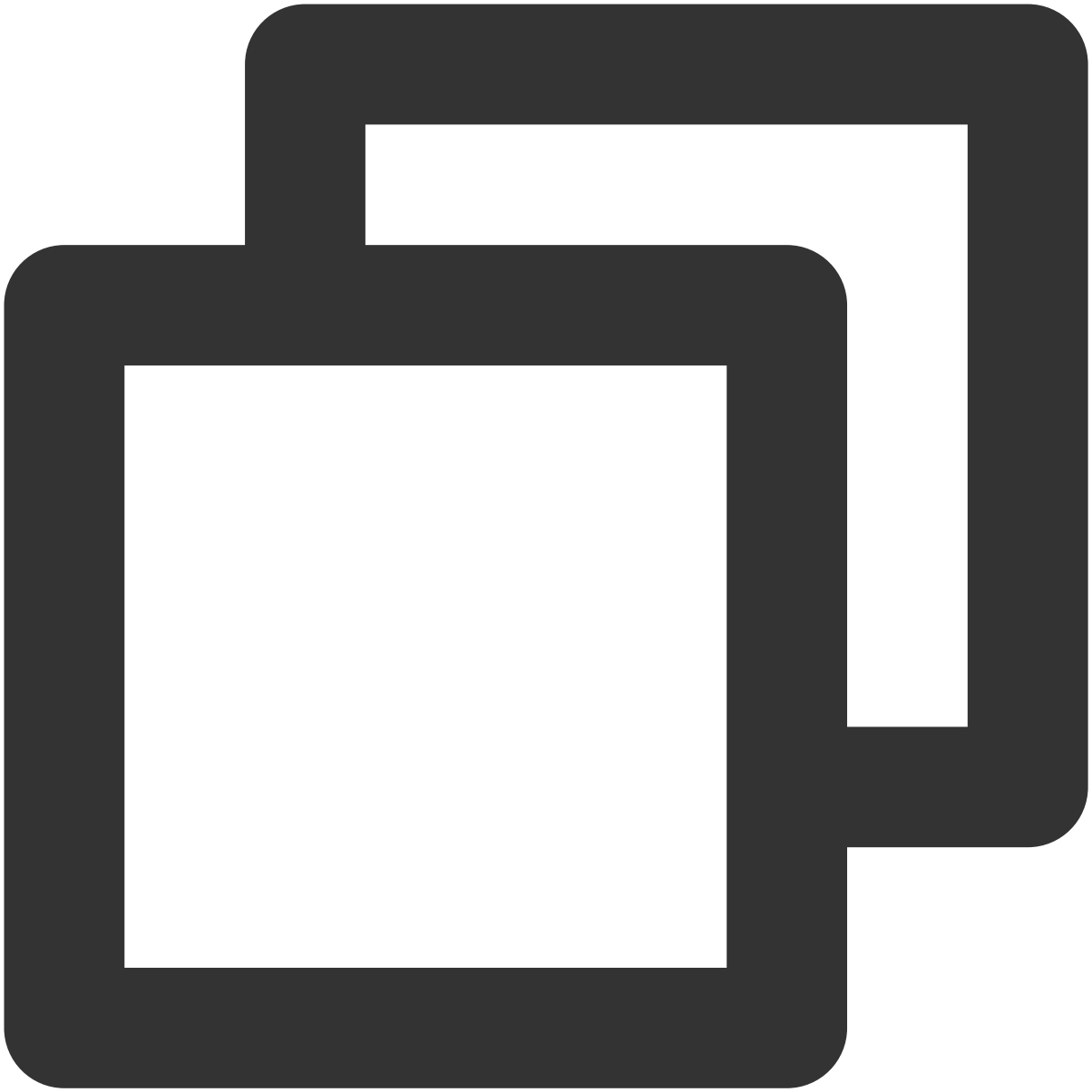
此时只能调用 Rest API 接口，如 App 运营人员希望给所有爱好足球的用户打上足球的标签，则可利用此接口进行批量绑定，反之进行批量解绑。

**标签绑定用法**



```
{
    "operator_type": 7,
    "tag_list": ["tag"],
    "token_list": ["token1","token2"]
}
```

**标签解绑用法**



```
{
    "operator_type": 8,
    "tag_list": ["tag"],
    "token_list": ["token1","token2"]
}
```

**使用限制**

每个标签最大50个字节。

每次调用设备 token 数量最大500个。

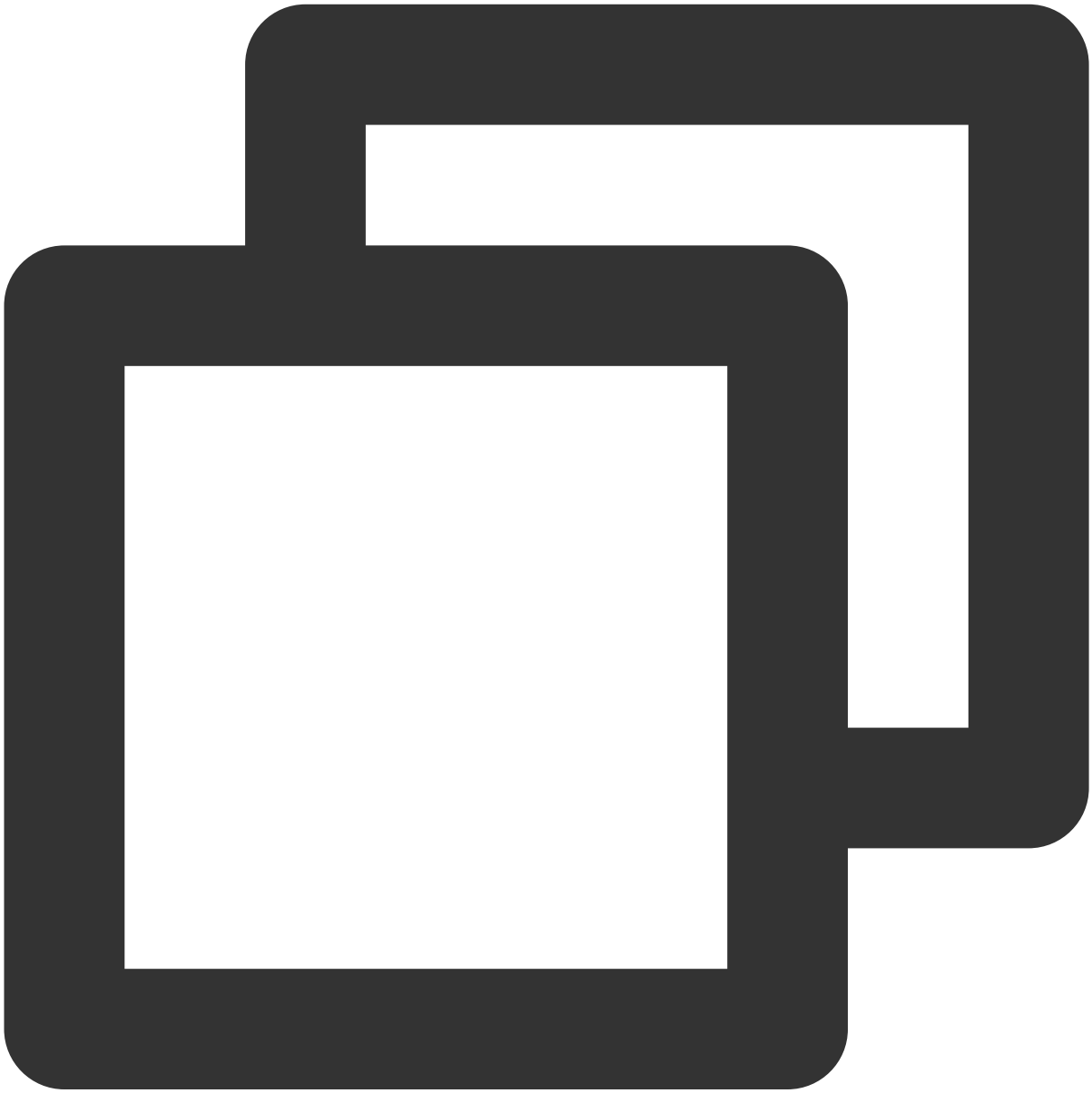
接口调用逻辑为同步。

**多标签多设备**

**推荐场景**

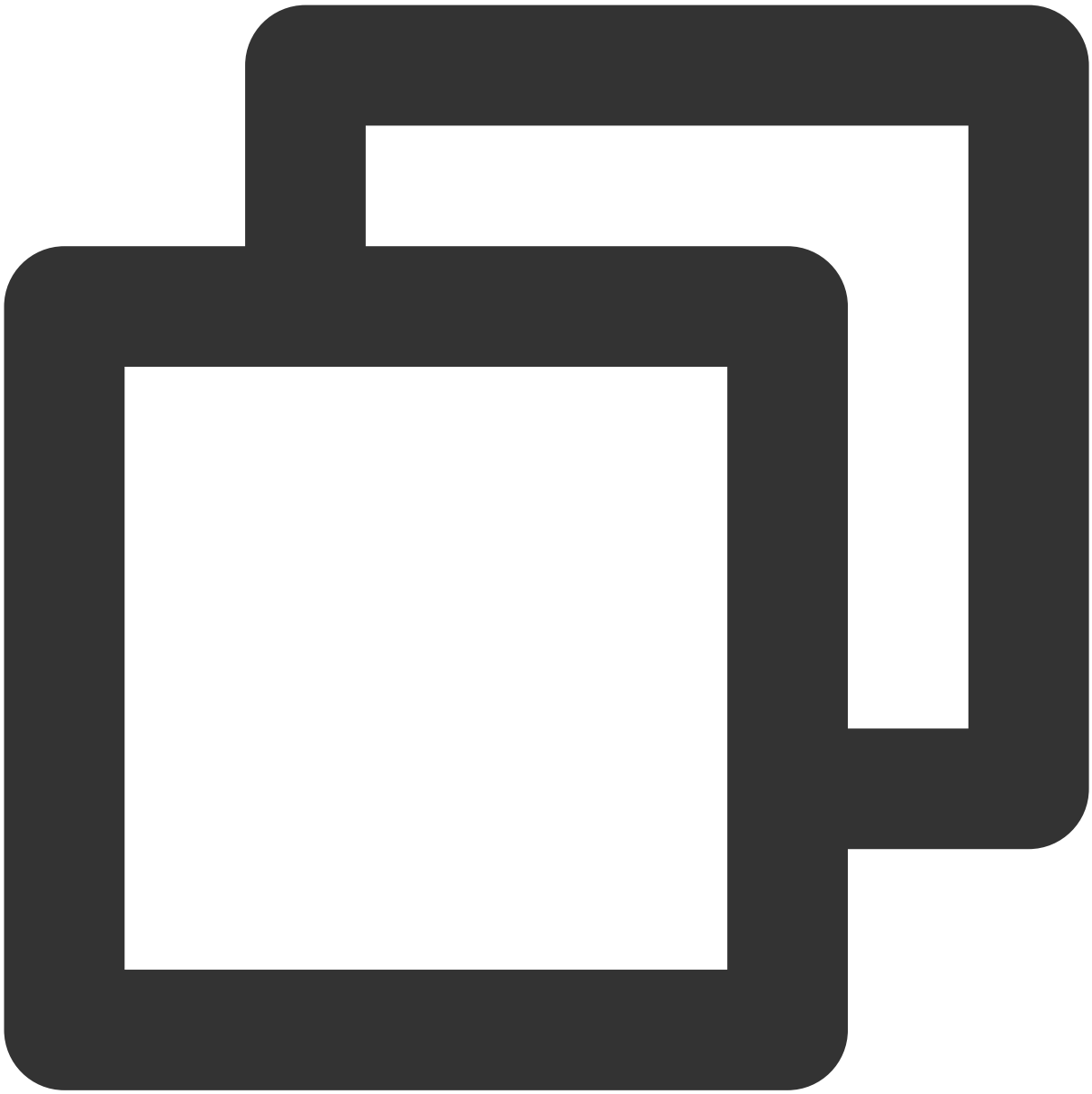
此时只能调用 Rest API 接口，如 App 运营人员希望给喜欢足球的用户打上足球的标签，喜欢篮球的用户打上篮球的标签，则可以利用此接口进行批量绑定，反之进行批量解绑。

**标签绑定用法**



```
{
    "operator_type": 9,
    "tag_token_list": [{"tag":"tag1","token":"token1"},{"tag":"tag2","token":"token
}
```

**标签解绑用法**



```
{
    "operator_type": 10,
    "tag_token_list": [{"tag":"tag1","token":"token1"},{"tag":"tag2","token":"token
}
```

**使用限制**

每个标签最大50个字节。

每次调用设备token数量最大500个。

接口调用逻辑为同步。

**标签覆盖**

移动推送提供了两种标签覆盖方式，一是普通覆盖，二是按标签类别进行覆盖，又名 kv 覆盖，此时":"号为关键字，作为 key-value 值的分割符。

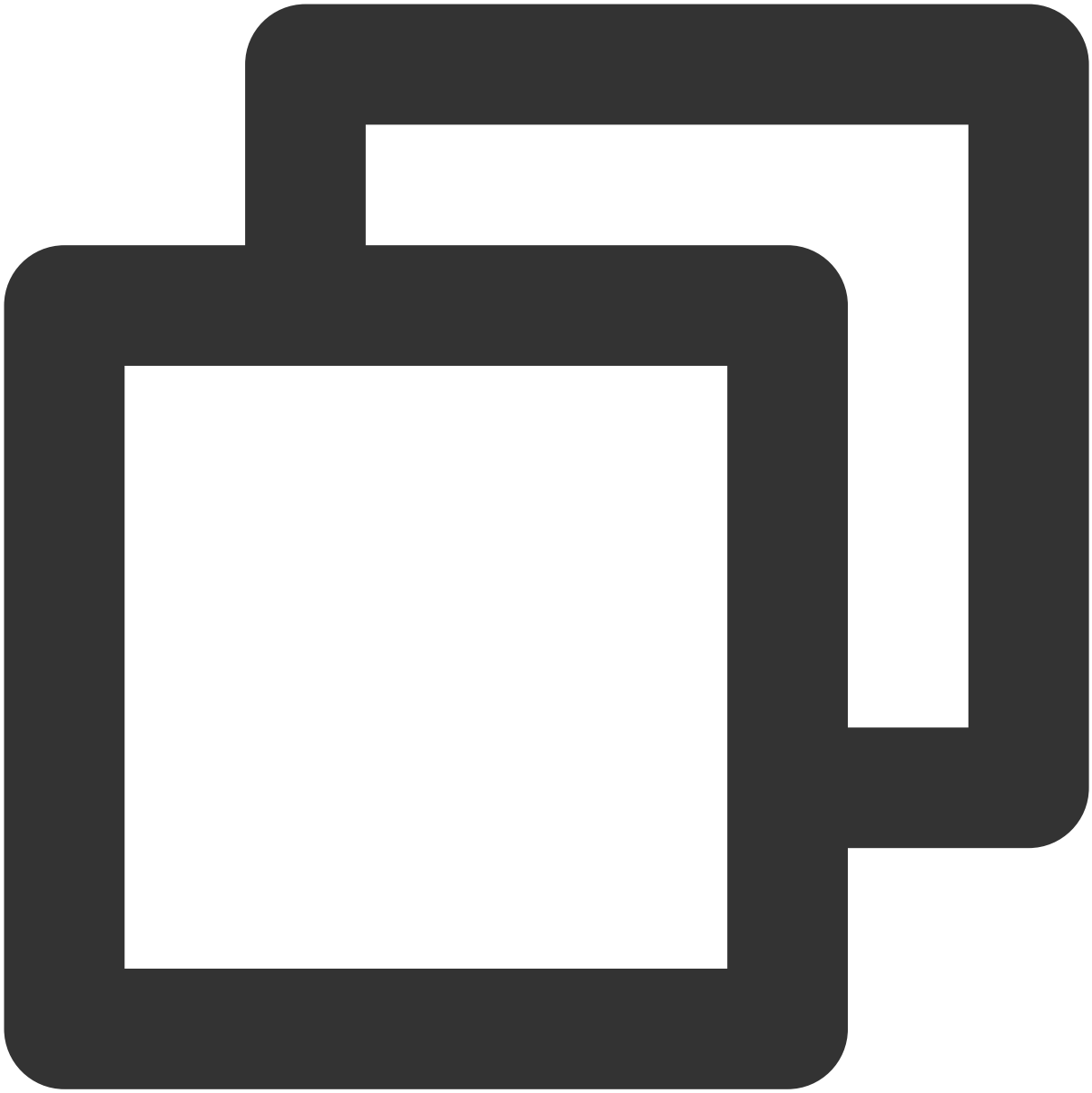
**普通覆盖**

**推荐场景**

1. 调用终端 SDK 接口，如该设备所订阅的所有频道信息已经失效，则需要对这些频道标签和设备进行一一解绑，但需将这些频道标签遍历出来再逐个解绑，则会很不方便，故此时可调用这个接口进行覆盖。
2. 调用 Rest API 接口，如 App 运营人员希望给设备重新设置标签，不想让这个设备受到之前标签的干扰，故可调用这个接口进行覆盖。

**标签覆盖用法**

版权所有：腾讯云计算（北京）有限责任公司



```
{
    "operator_type": 6,
    "tag_list": ["test", "level:1",, "level:2"],
    "token_list": ["token"]
}
```

**使用限制**

每个标签最大50个字节。

接口调用逻辑为异步，建议调用的频率间隔1秒以上。

**kv覆盖**

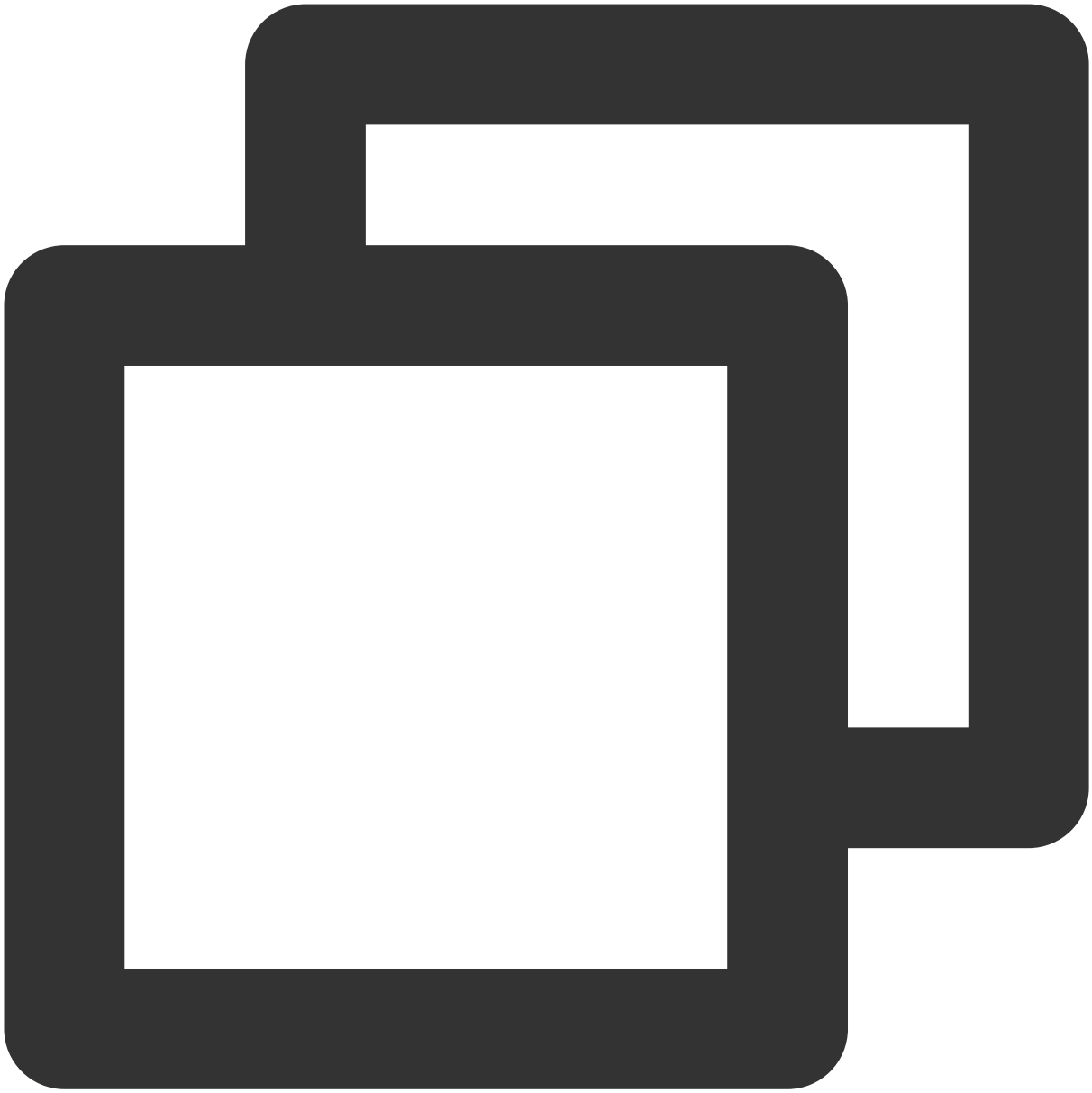
**推荐场景**

1. 调用终端 SDK 接口。

2. 调用 Rest API 接口。

**使用说明：**以":"进行分割，":"前作为标签类别，":"后面的作为这个标签具体的值，例如 level:2 表示该设备所在的等级为2级，假如这个设备对应的等级升级到3级，则需要先删除 level:2 标签，在添加 level:3 标签，如果客户明确知道这个设备仅有 level:2 这一个标签，则可调用普通覆盖接口进行覆盖，但往往一个设备会对应多个标签，如该设备还对应一个 test 标签，此时要精确删除 level:2 标签，就需要客户维护该设备对应的全部标签，或者调用移动推送接口把该设备对应的历史标签查找出来再进行删除，这就带来很多的不方便，故可调用此接口，仅进行 level 类标签的覆盖。

**标签覆盖用法**



```
{
    "operator_type": 6,
    "tag_list": ["test:2", "level:3"],
    "token_list": ["token"]
}
```

**接口说明：**当且仅当 tag_list 内全部标签都带有":"时，才会进行 kv 覆盖，例如该 token 已有 test,level:1标签，调用此接口后，该 token 的标签列表则变为 test, test:2, level:3。

**使用限制**

每个标签最大50个字节。

每次调用标签数量最大100个。
接口调用逻辑为异步，建议调用的频率间隔1秒以上。

**删除标签场景**

移动推送提供了两种删除标签的方式，一是删除单设备对应的全部标签，二是删除应用对应的某标签。

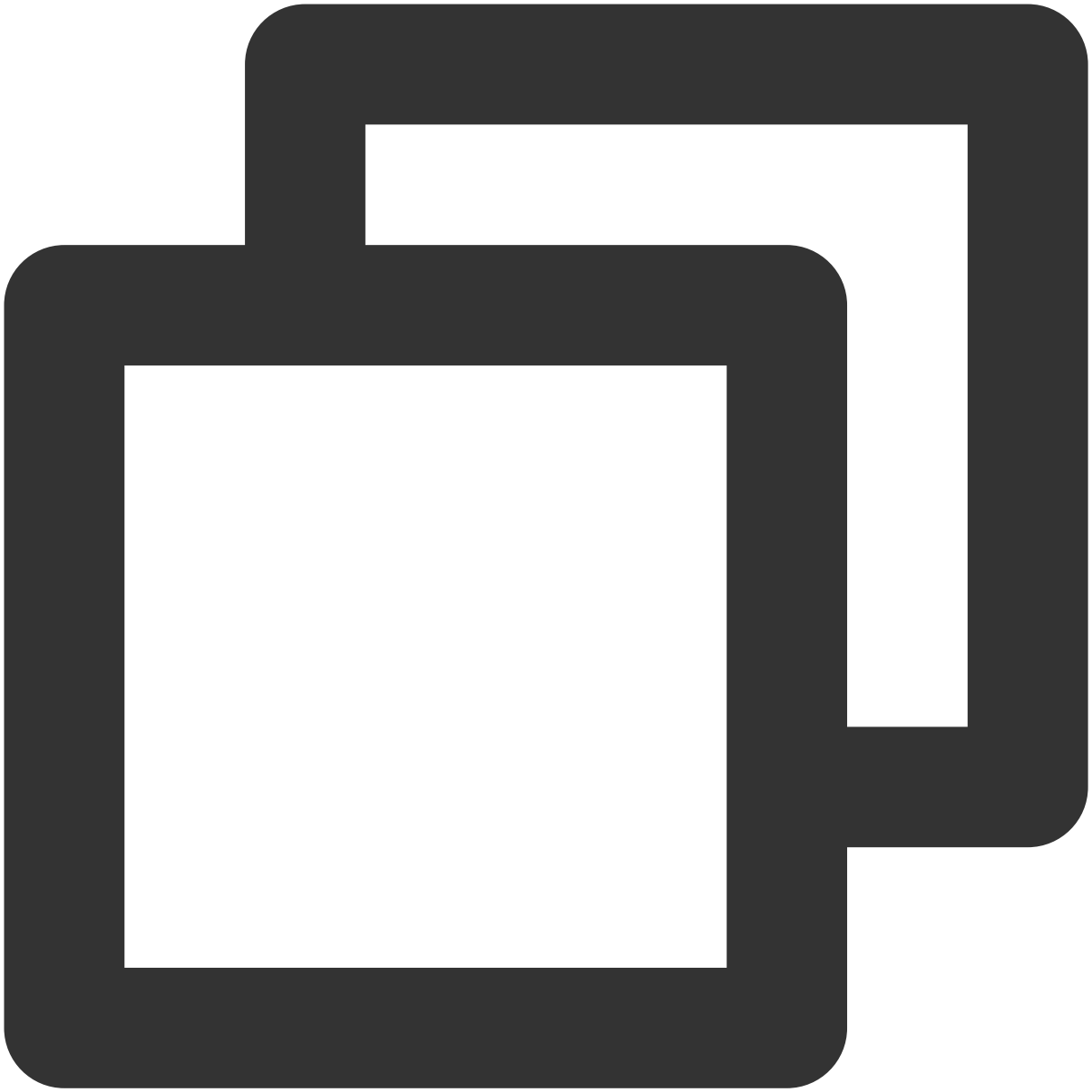
**删除单设备对应的全部标签**

**推荐场景**

1. 调用终端 SDK 接口。
2. 调用 Rest API 接口。

**使用说明：** 可利用此接口将设备对应的全部历史标签删除，一般针对设备对应的过期标签，或由于运营人员操作失误，希望重新对设备进行标签设置的场景。

**标签删除用法**



```
{
    "operator_type": 5,
    "token_list": ["token"]
}
```

**使用限制**

每个标签最大50个字节。

接口调用逻辑为异步，建议调用的频率间隔1秒以上。

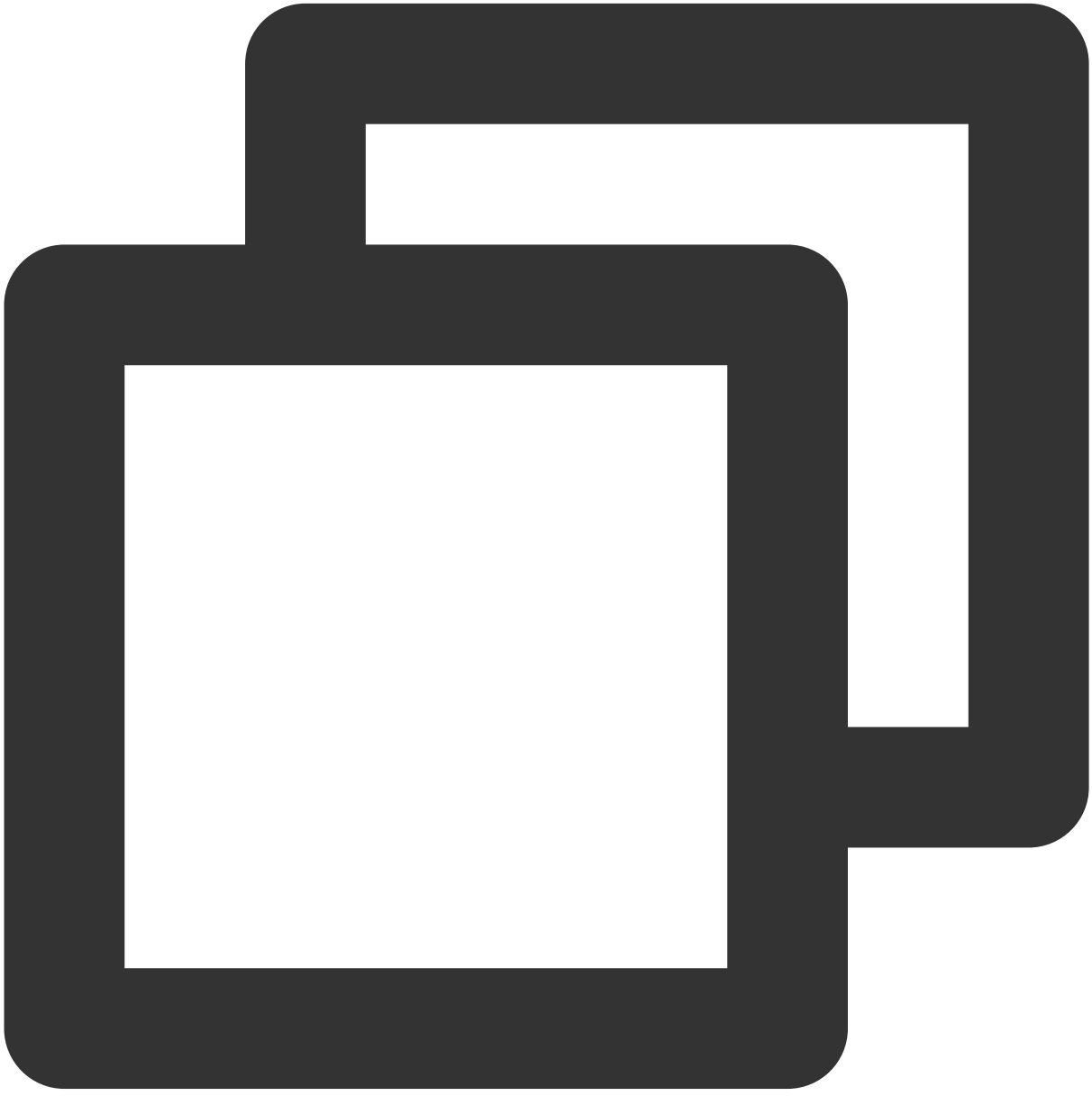
**删除应用对应的某标签**

**推荐场景**

（1）此时只能调用 Rest API 接口

**说明：**可利用此接口将应用对应的某些标签进行删除，即对这些标签与对应绑定的设备进行全部解绑之后，在从应用的标签列表中移除，一般应用在一些彻底废弃的标签，如测试时引入的一些test标签，在正式上线之后，将这些test标签移除。

**标签删除用法**



```
{
    "tag_list": ["tag1", "tag2"]
}
```

**使用限制**

每个标签最大50个字节。

每次调用标签数量最大100个。

接口调用逻辑为异步，建议调用的频率间隔1秒以上。

## 预设标签管理

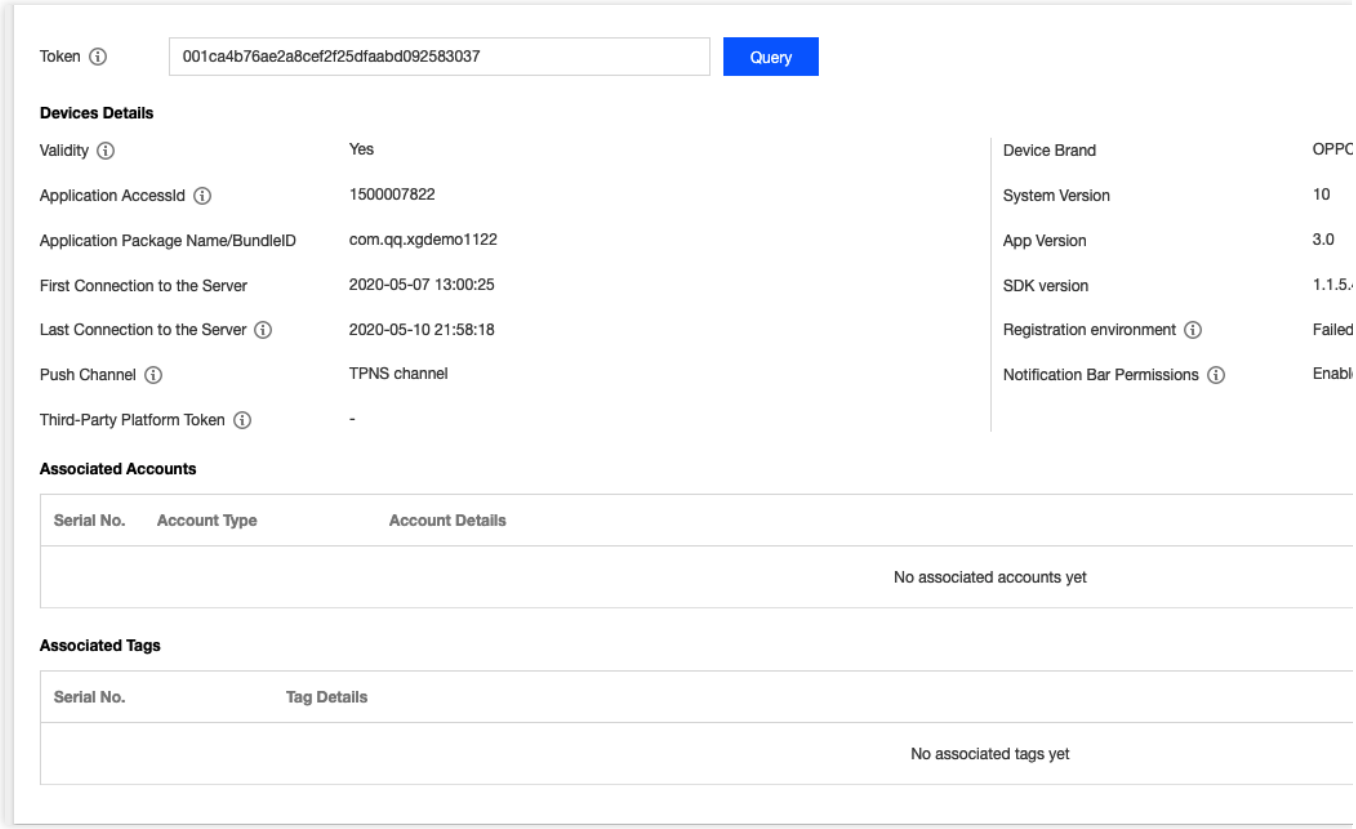
预设标签是指移动推送平台侧维护的标签，即在用户设备注册或连接到移动推送服务器时 SDK 自动采集的标签信息，目前移动推送平台维护的预设标签包括：App 版本，系统版本，省份，活跃信息，系统语言，SDK 版本，国家&地区，手机品牌，手机机型。

每个设备只会绑定最新的预设标签，并会进行自动互斥，如某设备的 App 版本当前为1.0.1，当这个设备的 App 版本升级到1.0.2时，则会自动将该设备和1.0.1版本标签进行解绑，并重新对该设备和1.0.2版本标签进行绑定。

## 设备绑定标签查询

可在腾讯云管理台的 **App 推送管理** > **排查工具**页面，根据设备 token 查询具体的预设标签和自定义标签。

如图所示：



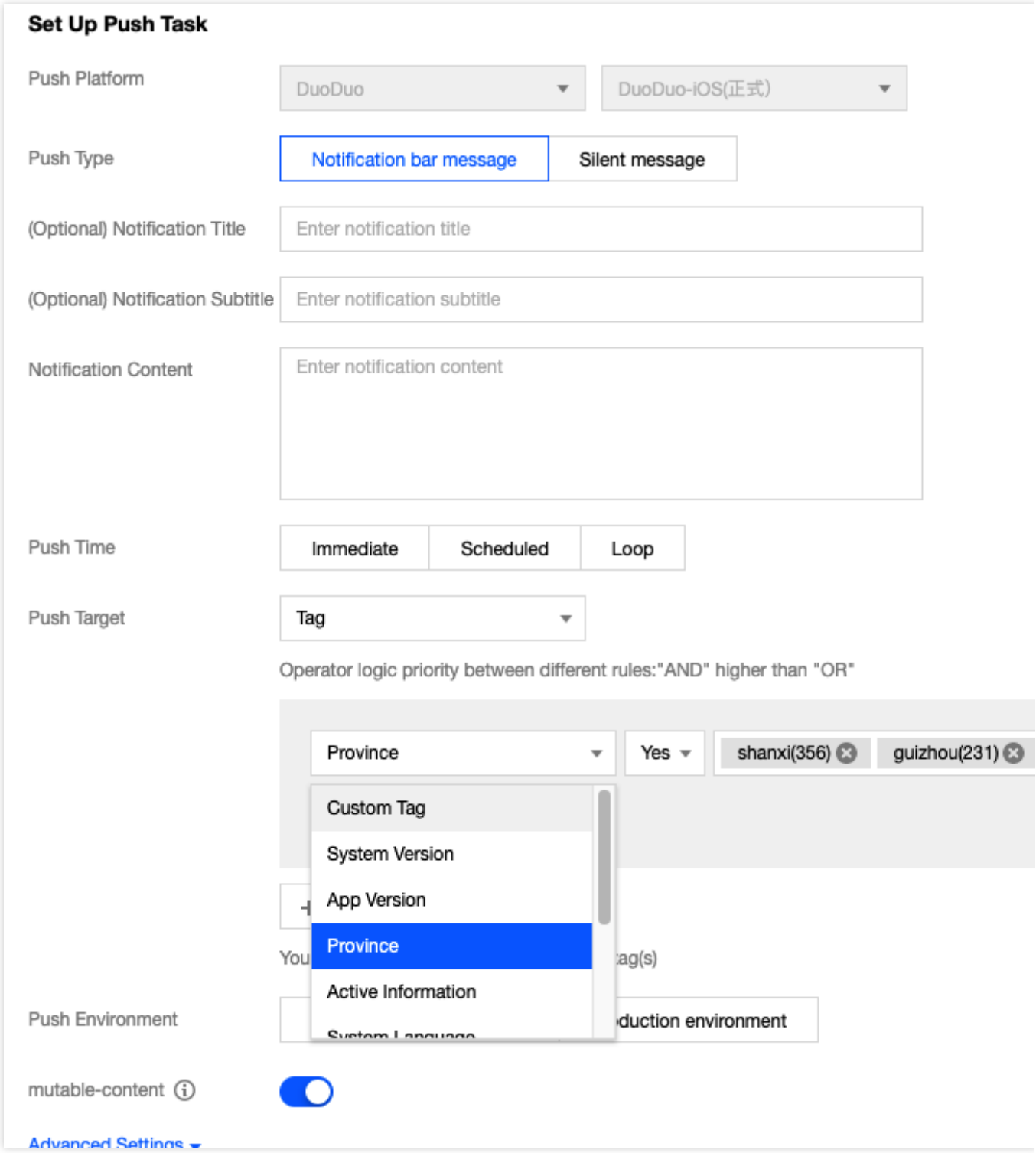

# 开始使用

标签推送支持使用预设标签或自定义标签进行推送，同时您也可以选择预设标签和自定义标签进行组合（与或组合）推送，多方面满足您的运营需求。

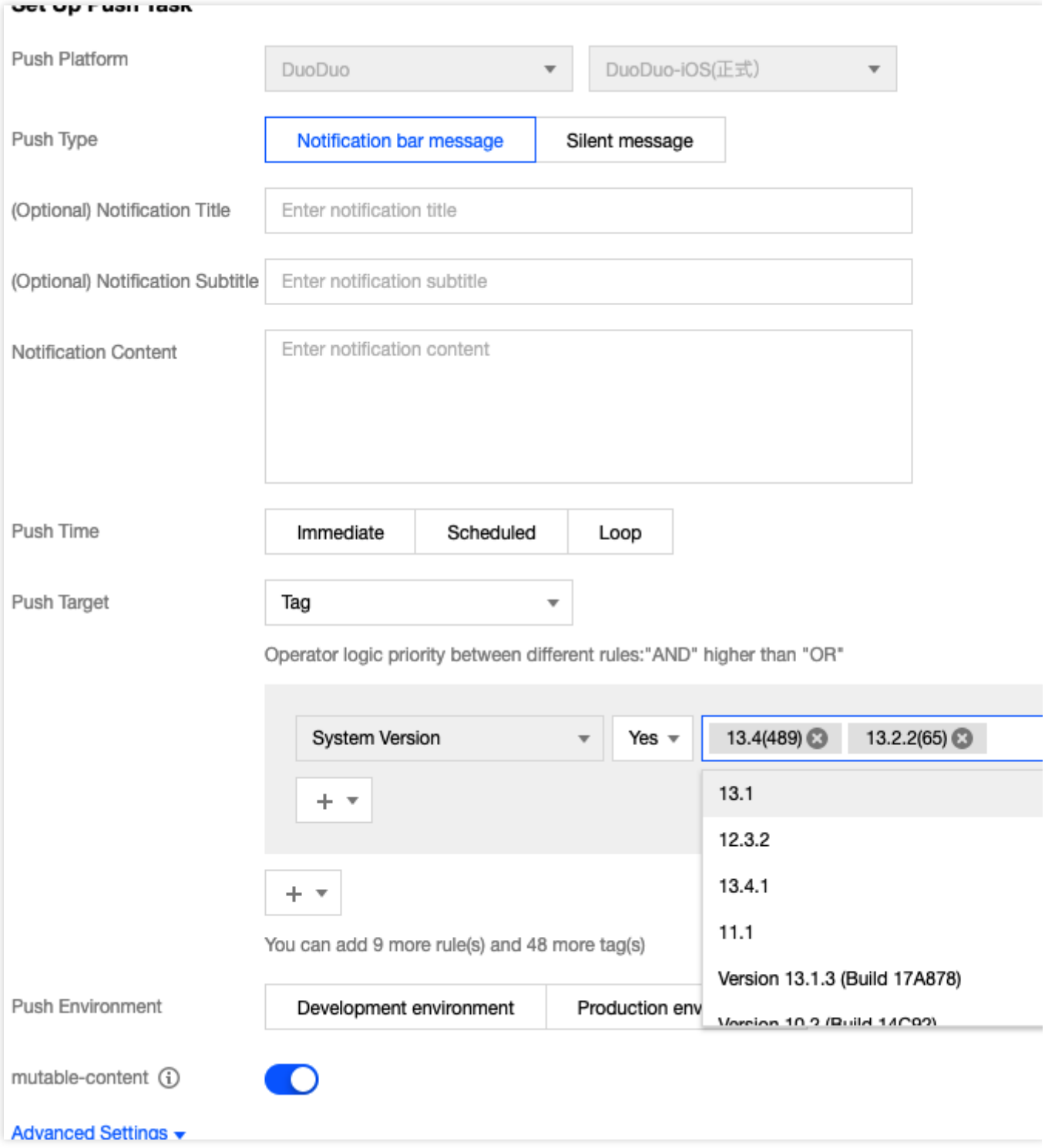
### 控制台使用

腾讯云管理台中进行标签推送，如下图所示步骤。

1. 先选择标签组合类型，可选自定义标签，和预设标签中的某类标签。



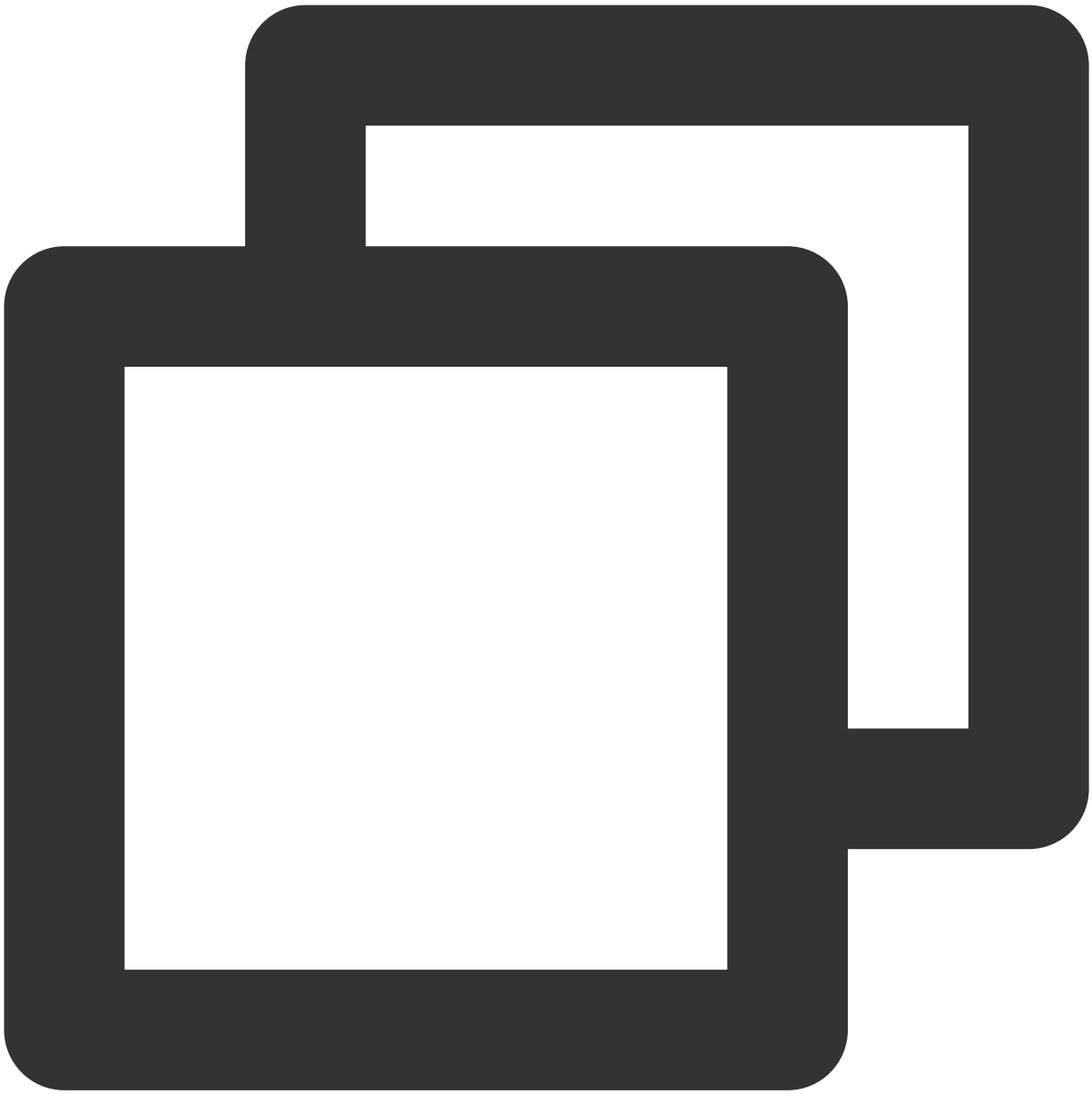
2. 在选定标签类型之后，可选择具体需要推送的标签。




选中标签之后，还会显示选中标签所对应的绑定设备数，上图表述的标签组合意思为：对广东且在2日内活跃过的用户进行推送。之后点击测试预览即可进行对应目标设备的推送。

### 调用 API 标签推送

在 Push API 请求参数中设置 audience_type（推送目标）为 tag， 即为标签推送，详情请参见 推送接口 文档。
API 示例:对广东和江苏的用户且在20200423，20200422和20200421这三天活跃过，并且是男性的用户进行推送



```
{
    "audience_type": "tag",
    "tag_rules": [
        {
            "tag_items": [
                {
                    "tags": [
                        "guangdong",
```

```
                        "jiangsu"
                    ],
                    "is_not": false,
                    "tags_operator": "OR",
                    "items_operator": "OR",
                    "tag_type": "xg_auto_province"
                },
                {
                    "tags": [
                        "20200421",
                        "20200422",
                        "20200423"
                    ],
                    "is_not": false,
                    "tags_operator": "OR",
                    "items_operator": "AND",
                    "tag_type": "xg_auto_active"
                }
            ],
            "operator": "OR",
            "is_not": false
        },
        {
            "tag_items": [
                {
                    "tags": [
                        "male"
                    ],
                    "is_not": false,
                    "tags_operator": "OR",
                    "items_operator": "OR",
                    "tag_type": "xg_user_define"
                }
            ],
            "operator": "AND",
            "is_not": false
        }
    ],
    "message_type": "notify",
    "message": {
    "title": "测试标题",
    "content": "测试内容",
    "android": {
        "ring": 1,
        "ring_raw": "ring",
        "vibrate": 1,
        "lights": 1,
```

```
          "clearable": 1,
          },
        "custom_content":"{\\"key\\":\\"value\\"}"
      }
  }
  }
```

# 常见问题

**1. 能否开放根据标签查绑定设备数据的 API 接口？**

此接口暂时内部使用，从保护系统稳定的角度出发，没有开放出去的计划，可在管理台推送页面上选中相应的标签后，查看对应绑定的设备数。

**2. 能否开放根据设备 token 查绑定标签的 API 接口？**

此接口暂时内部使用，从保护系统稳定的角度出发，没有开放出去的计划，可在管理台工具箱中输入对应 token 查询绑定的标签。

版权所有：腾讯云计算（北京）有限责任公司

# 多包名推送

最近更新时间：2024-01-16 17:34:39

对于各类 Android 应用，特别是 Android 游戏而言，针对不同的渠道定制版本、使用不同的包名是运营中常见的一环，但是这样会导致后续使用消息推送时工作量过大，每个包都要单独推送，费时费力，还难以做到精准推送，效率低下。
移动推送团队新推出的“多包名推送”就是针对这一现状的有利工具，开通多包名之后，就可以通过添加渠道包名来轻松添加各个渠道包的包名，一次消息推送就可以触及所有的渠道包。

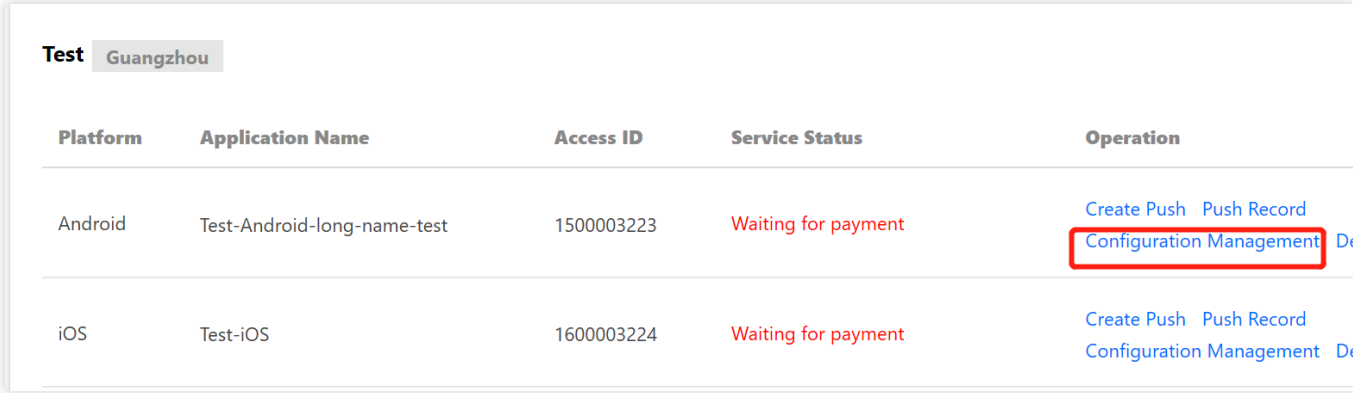
## 场景

某游戏因运营推广需求，需要发布一条活动公告给所有游戏玩家，该游戏上架了多个应用市场（例如酷安，安智、豌豆荚，360等），有不同的渠道包名，此时便可使用多包名推送功能，在发送推送消息时打开多包名推送开关，此时，所有该游戏的渠道包都可在同一时间收到此活动公告。

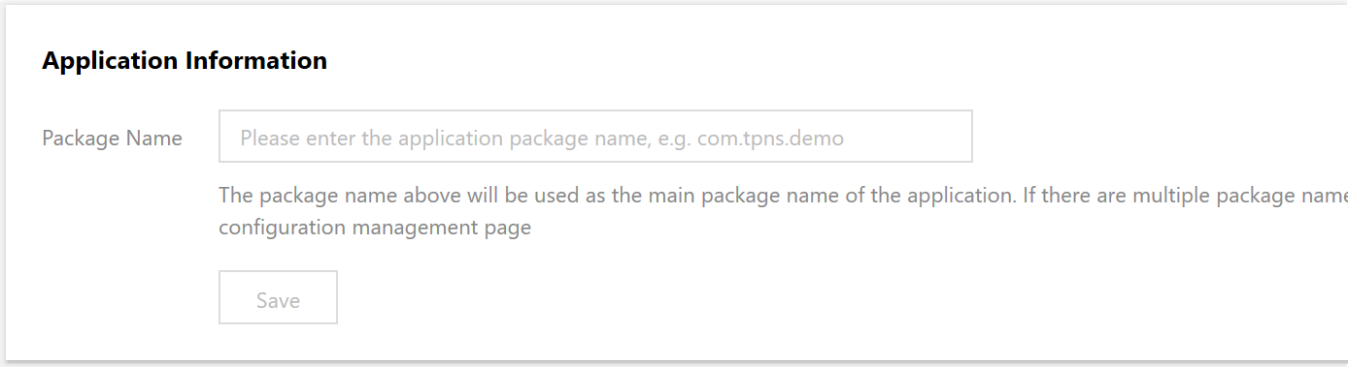
## 使用前准备

### 控制台配置

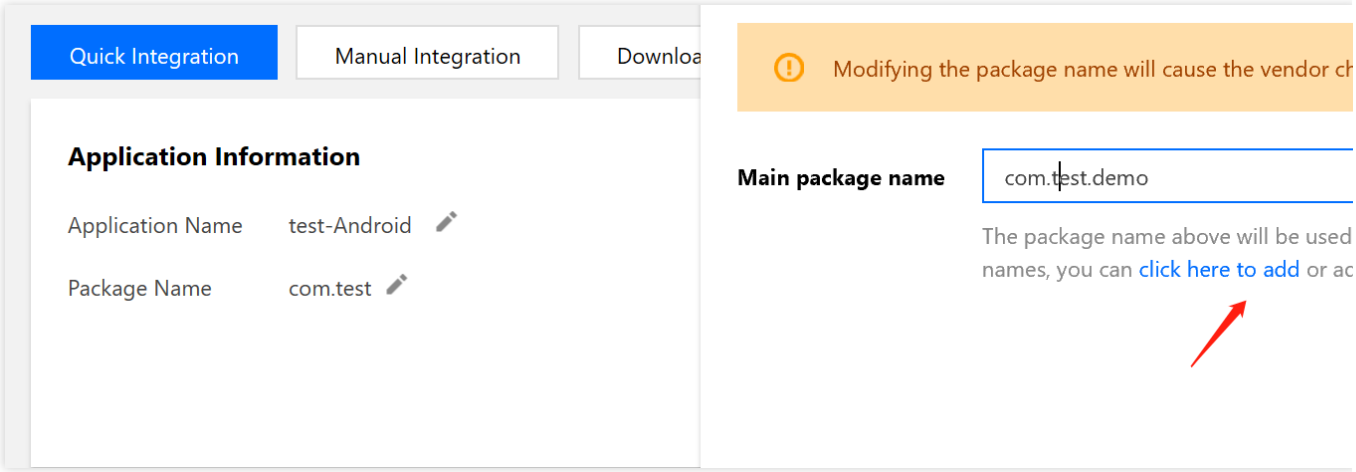
1. 进入移动推送控制台 产品管理页面。
2. 选择需要配置多包名的应用，单击**配置管理**。




3. 当应用未补充主包名时，需填入主包名同时单击**添加包名**，填入渠道包名。




4. 当应用已配置主包名可单击**编辑**进入包名管理，单击添加按钮补充渠道包名。



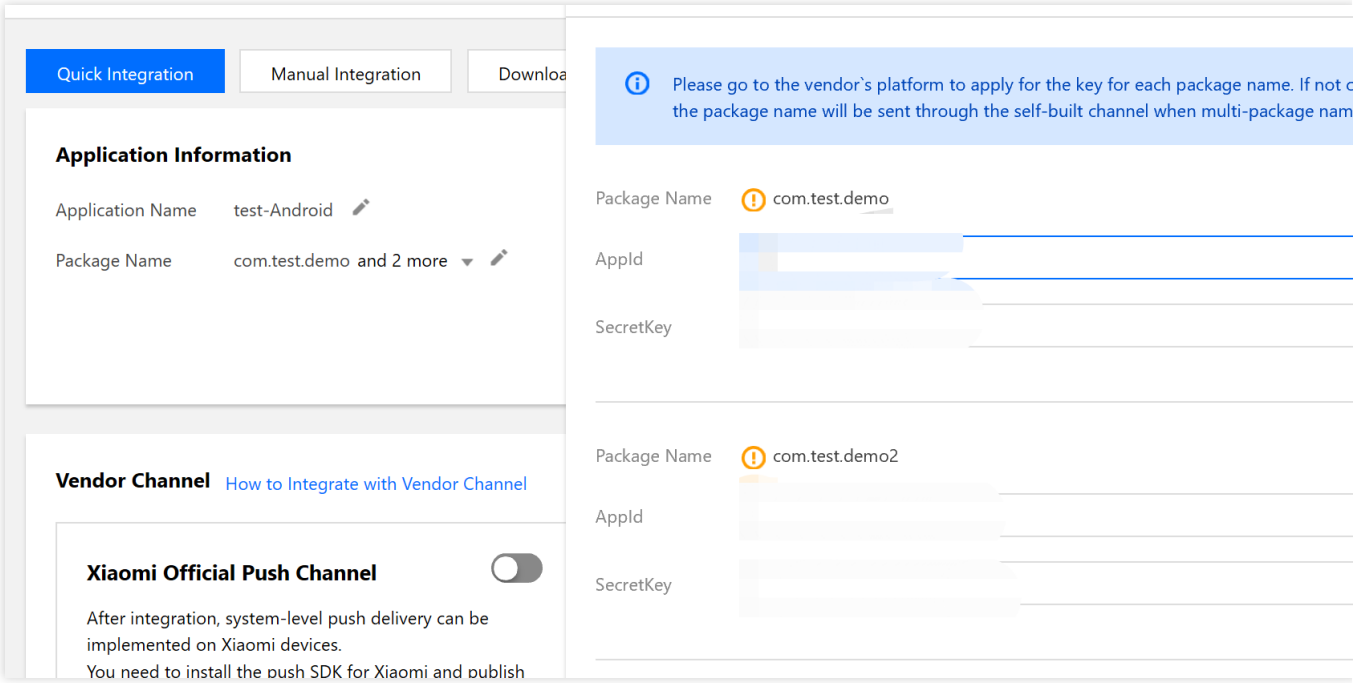

**注意：**

渠道包名最多支持配置50个。

### 多包名厂商通道配置

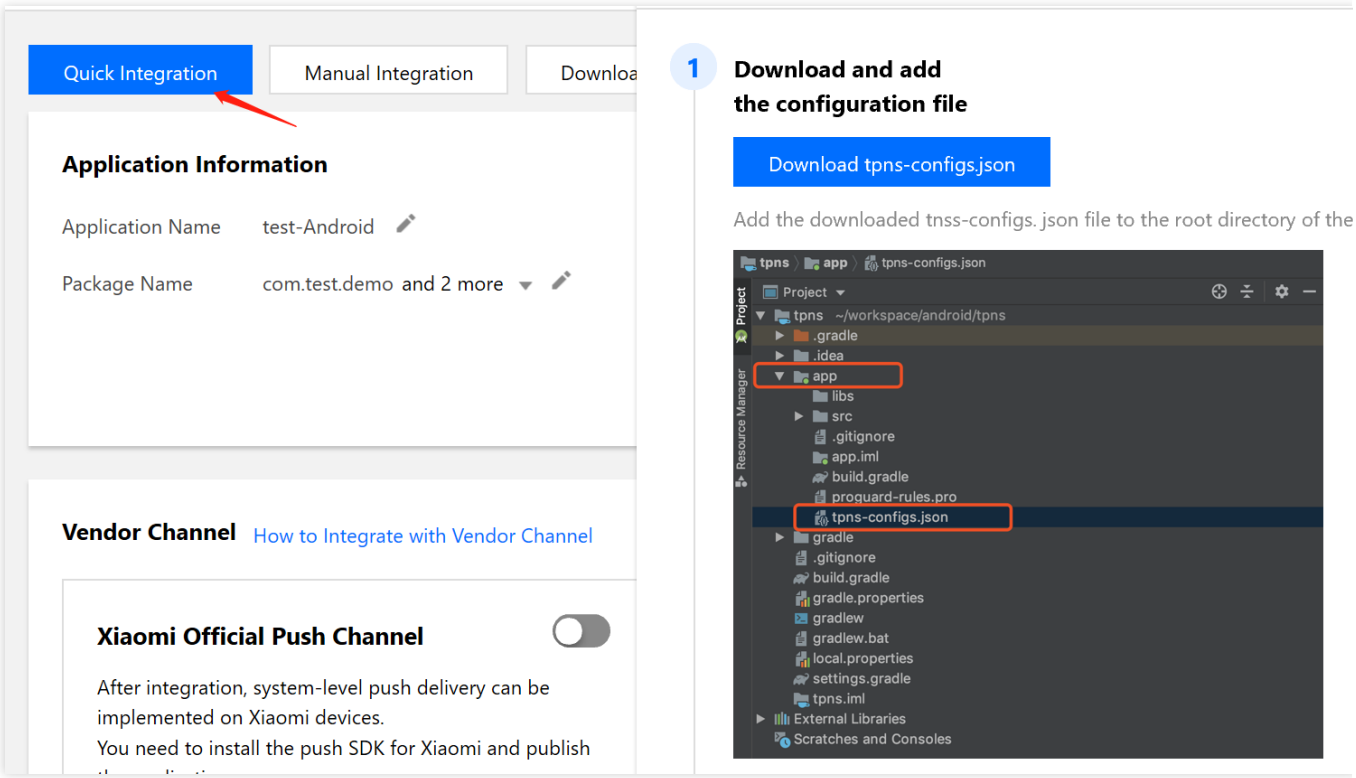
当应用有多个包名时，若需要通过厂商通道下发渠道包名的消息，则需要为每个渠道包名申请厂商密钥，并在 移动推送控制台 > **App推送管理** > **基础配置**页面进行配置。

以华为为例，当应用配置了多个包名时，开启华为通道将出现多套密钥配置，需将每个包名的配置补充完整，否则开启多包名推送时该包名下设备的消息将通过移动推送自建通道下发。




## 集成 SDK

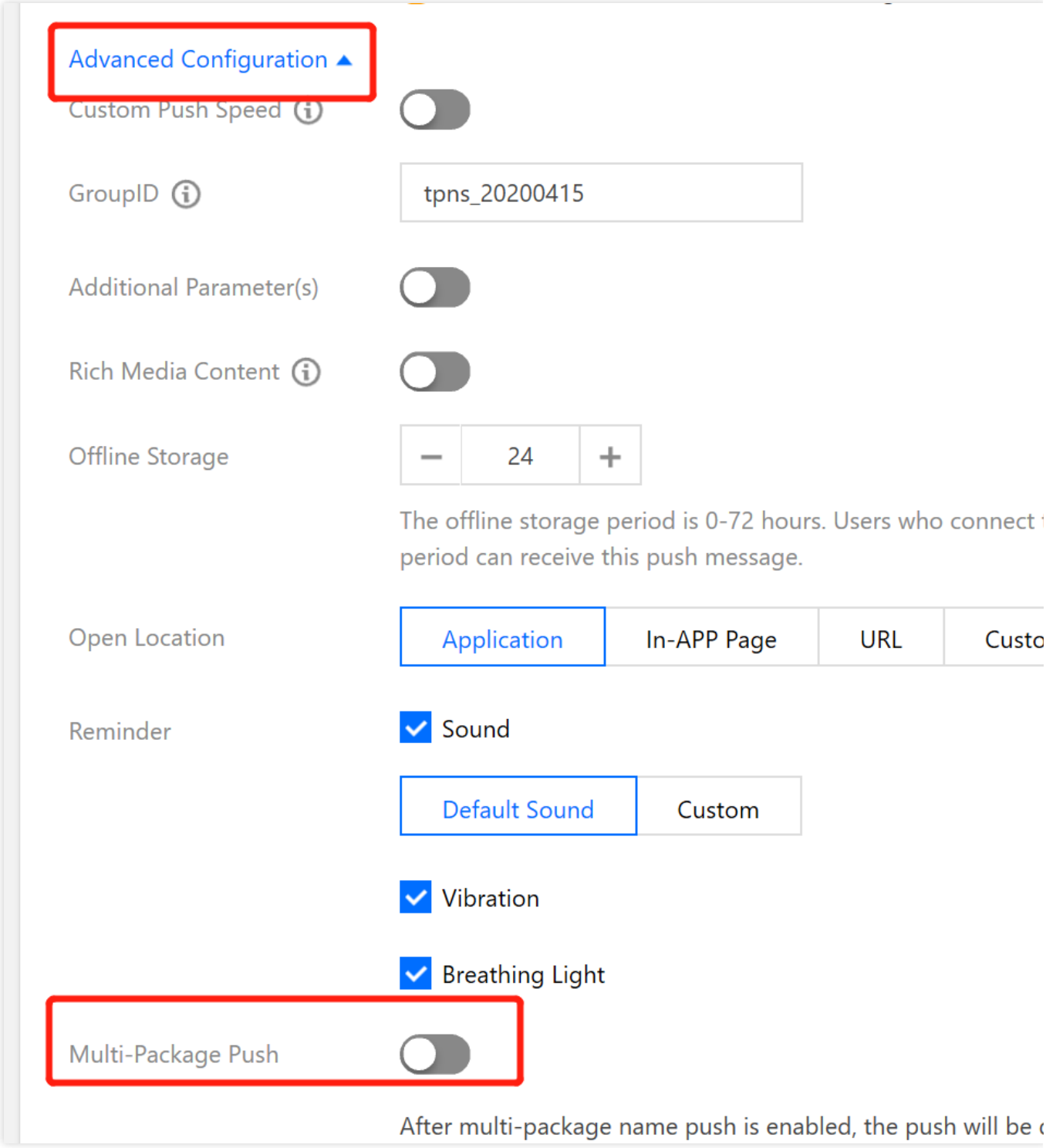
配置好包名之后，获取 `AccessID、AccessKey`，根据 Android SDK 集成指南 或按照控制台**快速接入**流程配置。



# 开始使用

### 控制台使用

当以上配置补充完整并确认无误后，可在**移动推送控制台 > 消息推送 > 新建推送 > 高级设置**多包名推送处开启多包名推送开关，如下图所示：

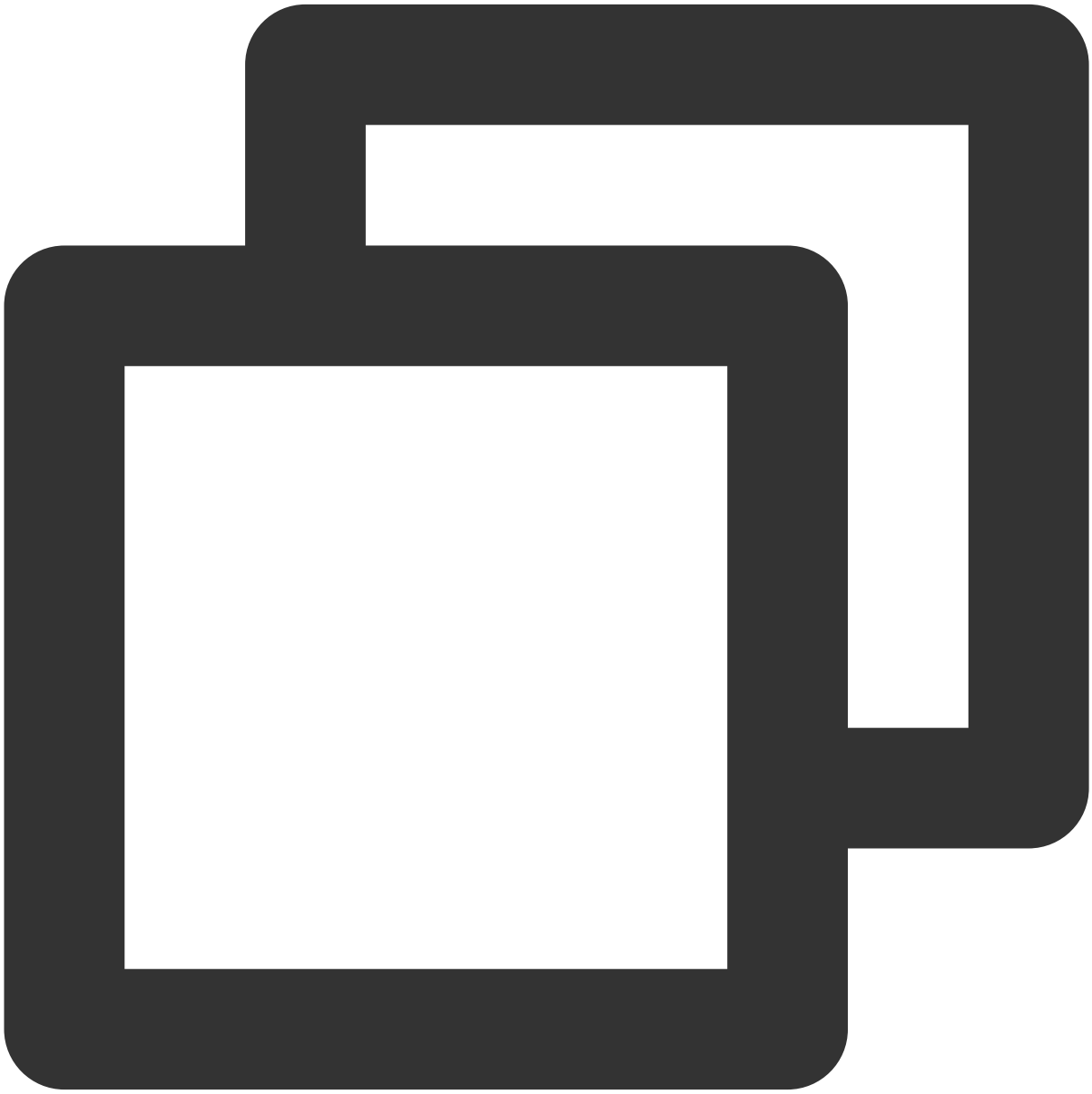



开启多包名推送后，推送将对所有包名下匹配推送目标的设备下发。

**说明：**
多包名推送功能仅限 Android 平台，若包名对应厂商通道未配置，则该包名下注册设备的消息通过移动推送自建通道下发。

### Rest API 使用

在 Rest API 可选参数中设置 `multi_pkg` 为 true 即为打开多包名推送，可参考 PushAPI 参数说明。
推送示例如下：



```
{
    "audience_type": "token",
```

```
    "token_list": [
        "05da87c0ae5973******9e08d884aada5bb2"
    ],
    "message_type": "notify",
    "multi_pkg":true,
    "message": {
        "title": "推送标题",
        "content": "推送内容",
        "android": {
            "custom_content":"{\\"key\\":\\"value\\"}"
        }
    }
}
```

# 通知点击跳转

最近更新时间：2024-01-16 17:34:39

## 使用场景

当订阅者点击您的通知时，可以根据您的配置跳转至指定的应用内页面、H5、Deeplink 等，来满足您在不同场景下的需求。

## 适用范围

| 平台 | 支持类型 |
| --- | --- |
| Android | Intent 方式跳转：点击通知可以跳转至指定应用内页面，并且可以携带自定义参数<br>打开应用：点击通知可以直接进入 App 主页面<br>URL：点击通知可以打开浏览器进入指定的网页<br>应用内 Activity：点击通知可以跳转至指定应用内页面 |
| iOS | 点击通知默认是打开 App<br>根据下发的自定义 key 和 value 来做相应业务逻辑 |

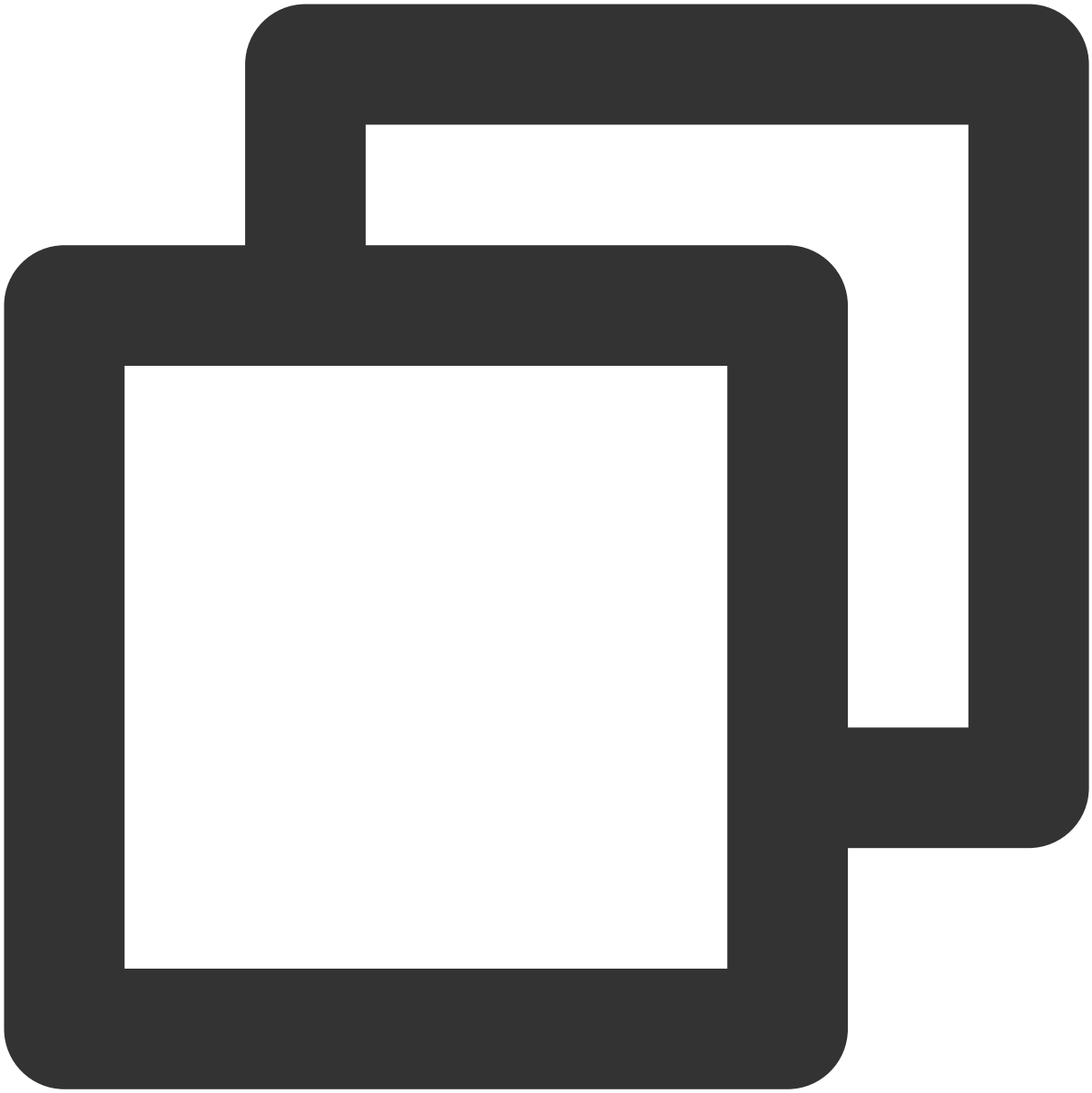
## Android 使用

**注意：**

SDK 点击消息默认支持点击事件，触发后打开目标界面，如果在 onNotifactionClickedResult 设置跳转操作会与管理台/ API 中指定的自定义跳转冲突，导致自定义的跳转失效。

### SDK 配置

如果使用 Intent 方式跳转，请首先在客户端 App 的 AndroidManifest 文件内，配置需要跳转的页面：
如需跳转 AboutActivity 指定页面，示例代码如下：



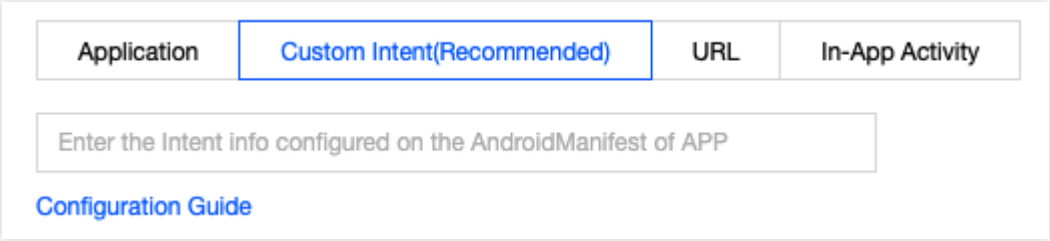
```
<activity
    android:name="com.qq.xg.AboutActivity"
    android:theme="@android:style/Theme.NoTitleBar.Fullscreen" >

    <!-- 其他 intent-filter -->
    <!-- <intent-filter> ... </intent-filter> -->

    <!-- AndroidManifest 支持为一个安卓组件配置多个 intent-filter，请将自定义跳转配置添加在
    <intent-filter >
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.DEFAULT"/>
```

```
        <!-- 自定义 data 块内容以指定您的完整 scheme -->
        <!-- 按照您的配置，将会组成形如"语义名://主机名/路径名"的 url 标识 -->
        <!-- 建议使用带有 app 名称、包名等可以唯一标记应用的字段进行配置，防止和其他应用的跳转目
        <data
              android:scheme="语义名"
              android:host="主机名"
              android:path="/路径名" />
    </intent-filter>
</activity>
```

## 控制台使用

### 使用 Intent 方式跳转（推荐）

使用 移动推送控制台 设置 Intent 进行跳转，在**推送任务 > 新建推送 > 高级设置 > 点击打开**中填写如下：




### 打开应用

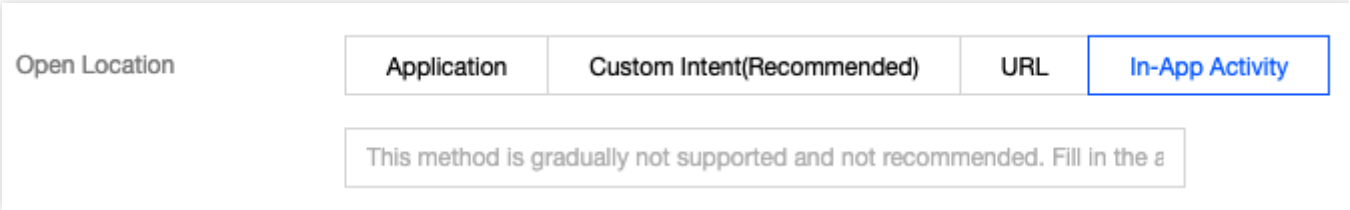
控制台推送默认是选择打开应用。

### 跳转至 URL

使用 移动推送控制台 设置 URL 进行跳转，在**高级设置 > 点击打开**中填写如下：

### 跳转至应用内 Activity

该方式后续将下线，不建议使用。在**高级设置 > 点击打开**中填写如下：



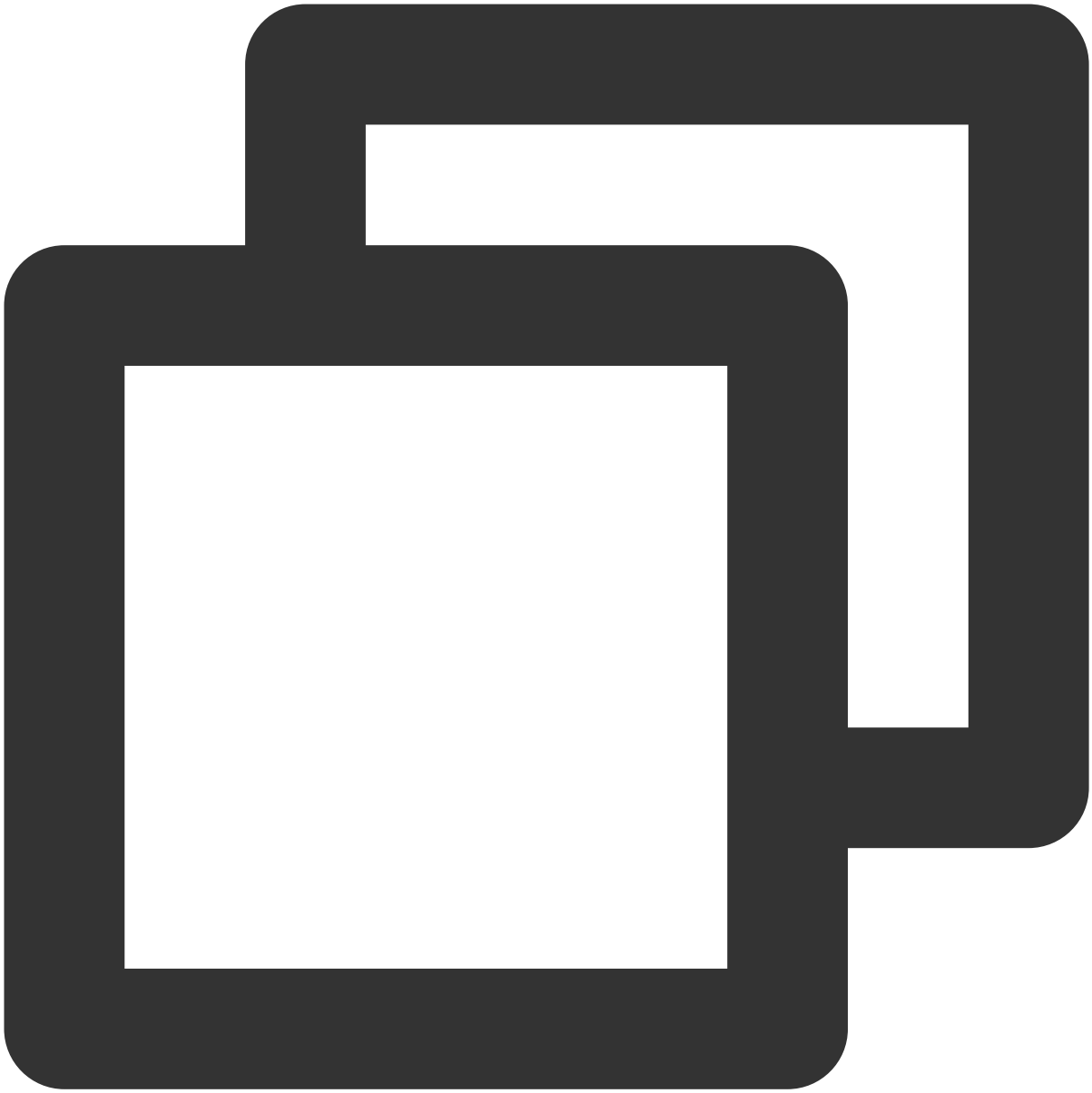

## Rest API 使用

在推送消息体中，`body.message.android` 下添加 `action` 和 `action_type` 字段，属性如下：

| 字段名 | 类型 | 父项目 | 默认值 | 必需 | 参数描述 |
| --- | --- | --- | --- | --- | --- |
| action | Object | Android | 有 | 否 | 设置点击通知栏之后的行为，默认为打开 App |
| action_type | Integer | Action | 有 | 否 | 点击动作类型，<br>1：打开 activity 或 App 本身<br>2：打开浏览器<br>3：打开 Intent（推荐 配置指引） |

### 使用 Intent 方式跳转（推荐）

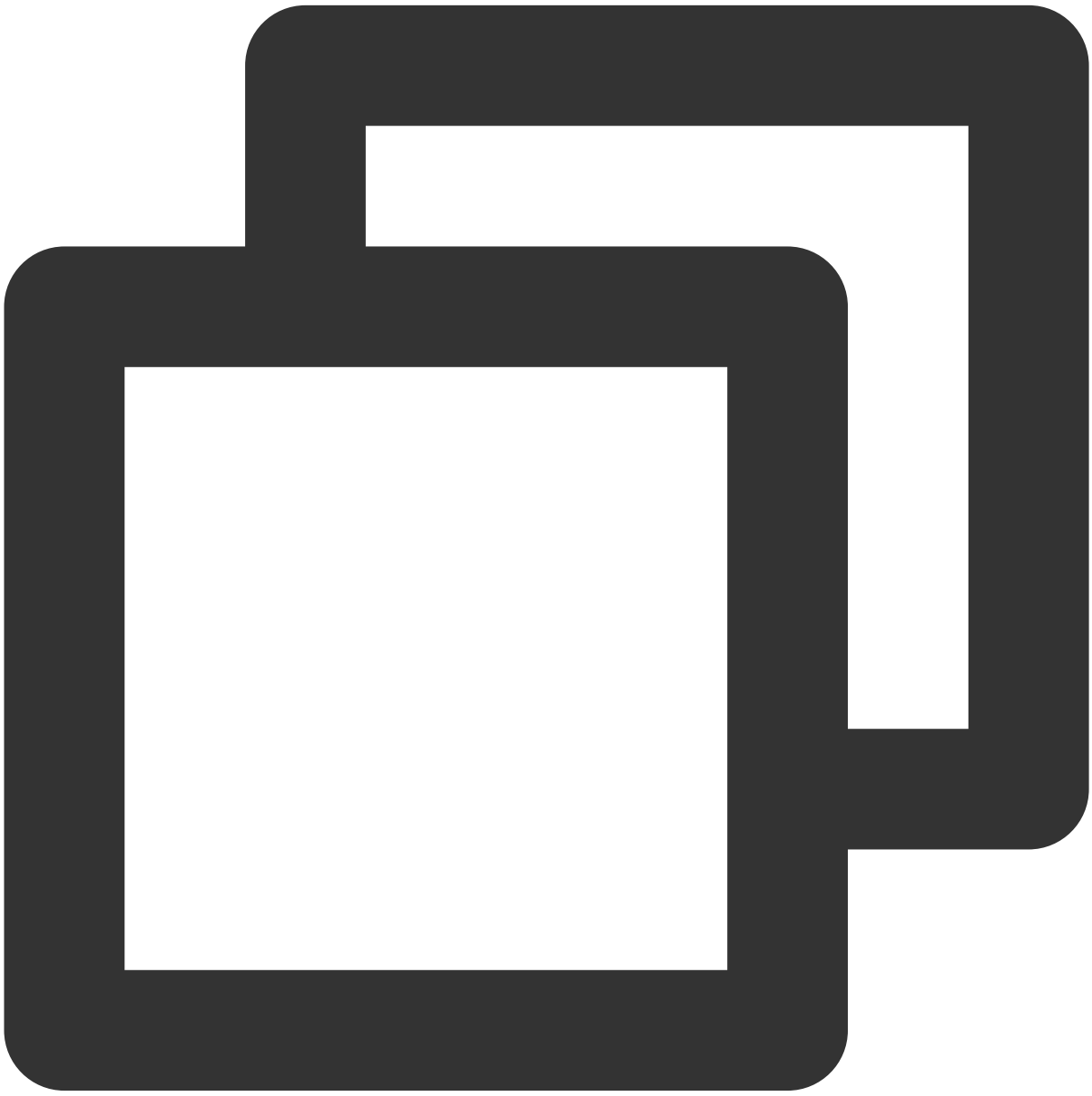
完整的消息示例如下：



```
{
  "audience_type": "token",
  "token_list": [
      "04xxx993"
  ],
   "message_type": "notify",
    "message":{
    "title": "xxx",
    "content": "xxx",
    "android": {
      "action": {
```

```
            "action_type": 3, // 动作类型，1，打开 activity 或 App 本身；2，打开浏览器；3，
            "intent": "xgscheme://com.tpns.push/notify_detail" //SDK 版本需要大于等于1
        }
      }
    }
}
```

若需要带上 param1 和 param2 等业务自定义参数，您可以做如下设置：



```
{
  "audience_type": "token",
  "token_list": [
```

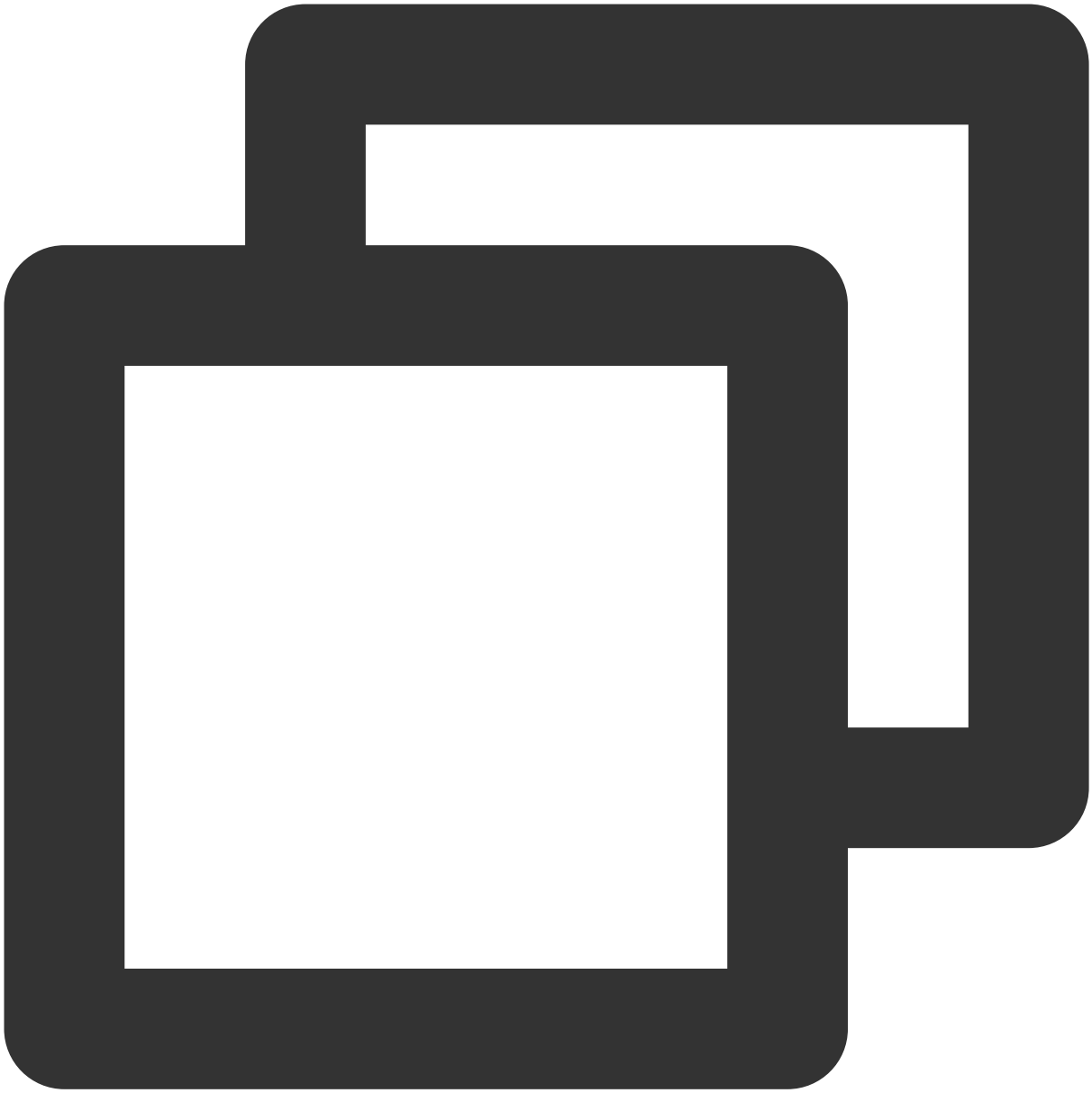
```
        "04xxx993"
    ],
     "message_type": "notify",
      "message":{
      "title": "xxx",
      "content": "xxx",
      "android": {
        "action": {
              "action_type": 3, // 动作类型，1:打开 activity 或 App 本身；2:打开浏览器；3:打
              "intent": "xgscheme://com.tpns.push/notify_detail?param1=aa&param2=bb"
          }
        }
     }
}
```

**说明：**

如果客户端需要获取参数做其他响应，参考 客户端获取参数。

**打开应用**

完整的消息示例如下：

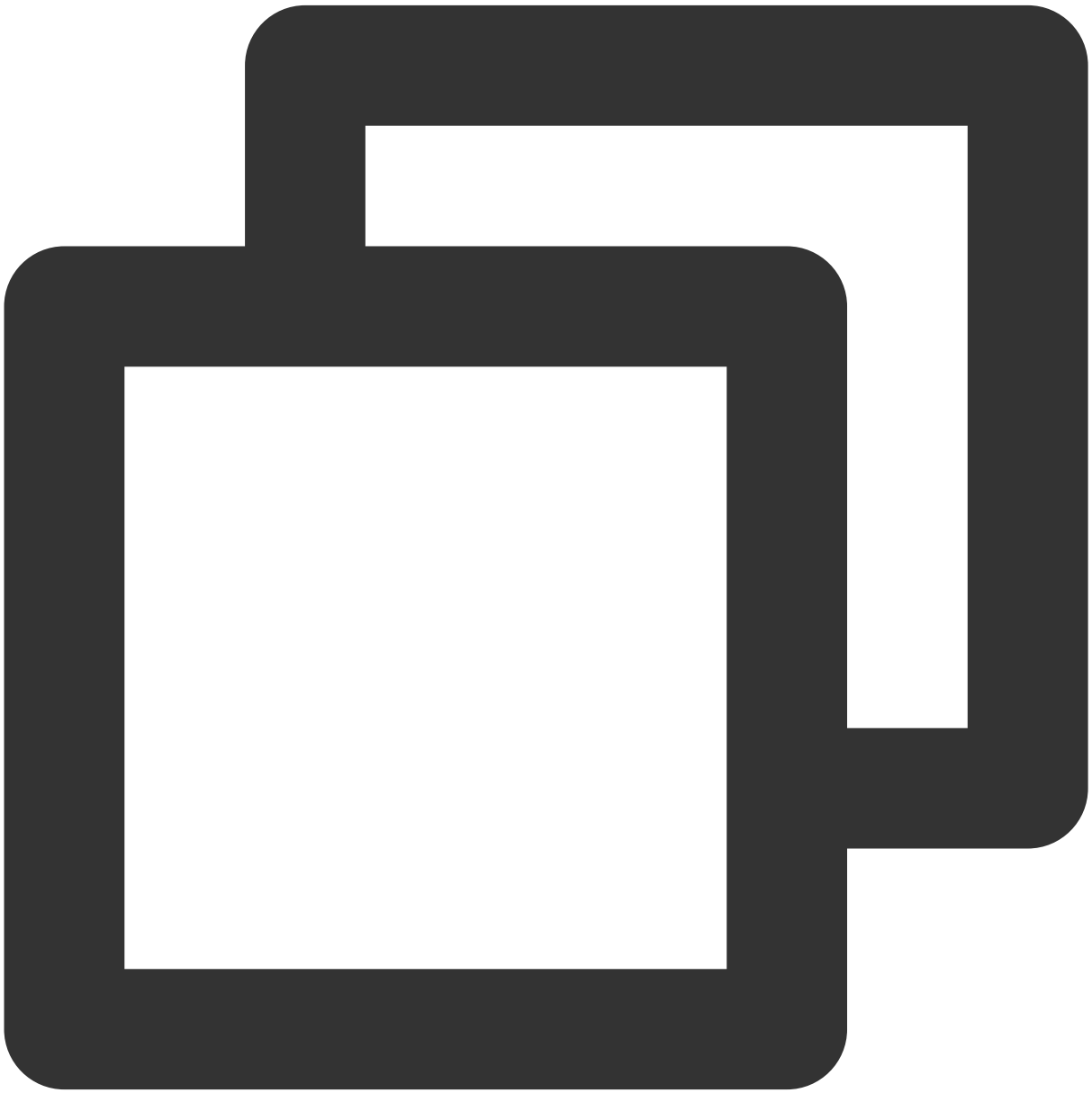


```
{
  "audience_type": "token",
  "token_list": [
      "04xxx993"
  ],
   "message_type": "notify",
    "message":{
    "title": "xxx",
    "content": "xxx",
    "android": {
      "action": {
```

```
            "action_type": 1 // 动作类型, 1:打开 activity 或 App 本身；2:打开浏览器；3:打开
          }
        }
    }
}
```

**跳转至 URL**

完整的消息示例如下：

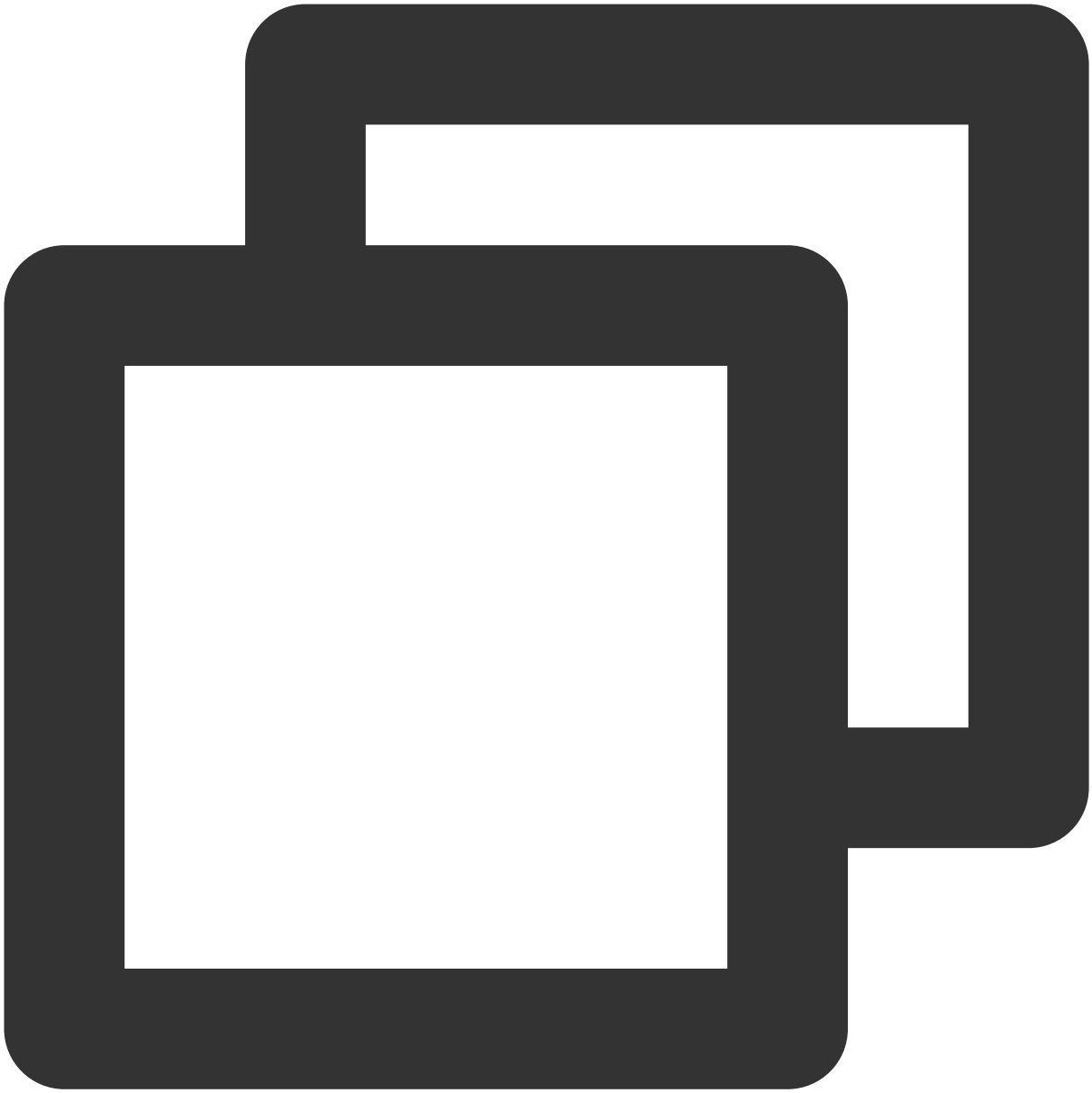


```
{
```

```
   "audience_type": "token",
   "token_list": [
       "04xxx993"
   ],
    "message_type": "notify",
     "message":{
     "title": "xxx",
     "content": "xxx",
     "android": {
       "action": {
            "action_type": 2, // 动作类型，1:打开 activity 或 App 本身；2:打开浏览器；3:打
            "browser": {
                 "url": "http://tpns.qq.com", // 仅支持http、https
                 "confirm": 1 // 是否需要用户确认
             }
         }
       }
    }
}
```

**打开应用内 Activity**

完整的消息示例如下：

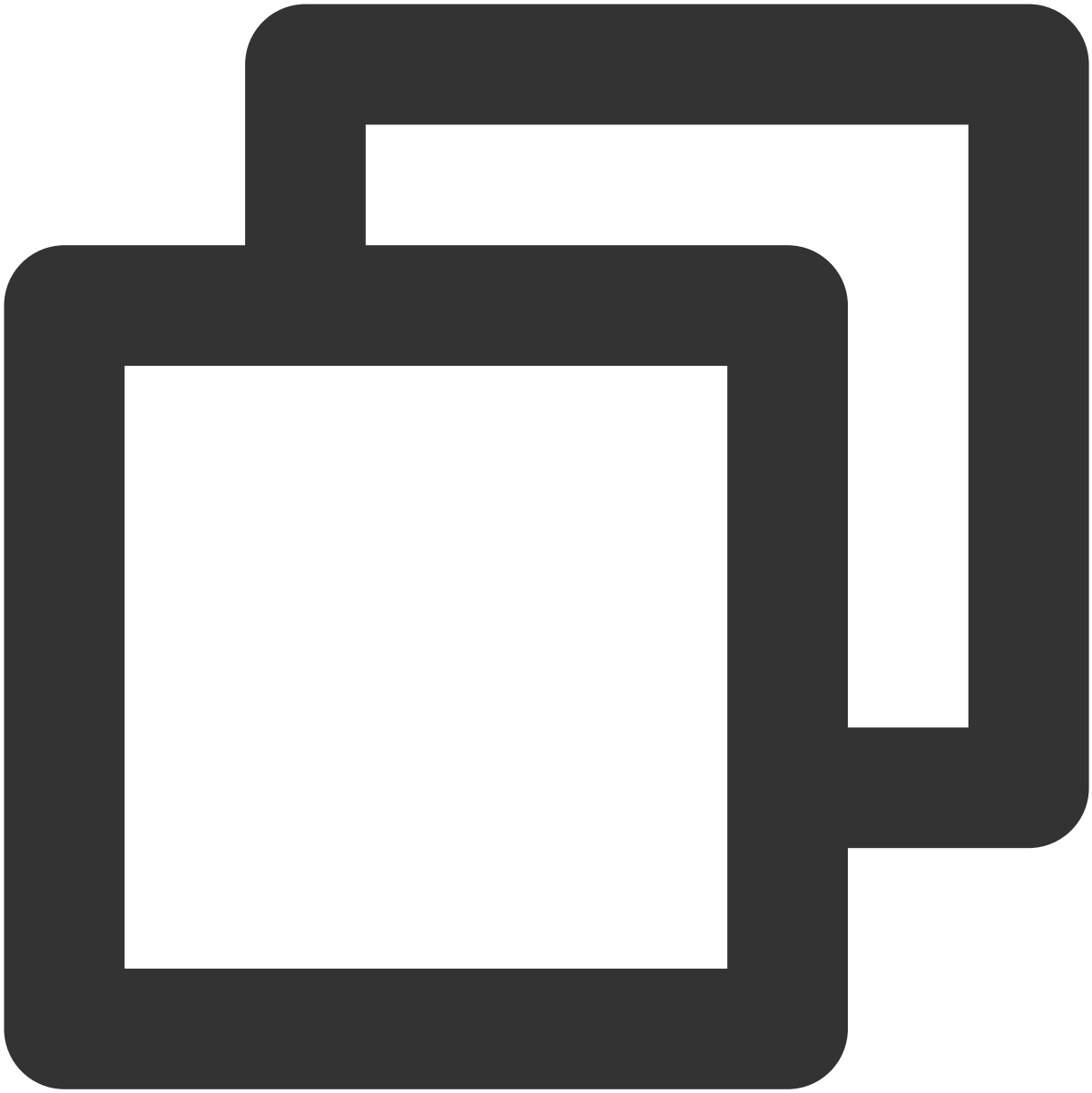


```
{
  "audience_type": "token",
  "token_list": [
      "04xxx993"
  ],
   "message_type": "notify",
    "message":{
    "title": "xxx",
    "content": "xxx",
    "android": {
      "action": {
```

```
            "action_type": 1, // 动作类型，1:打开 activity 或 App 本身；2:打开浏览器；3:打
            "activity": "com.x.y.MainActivity",
            "aty_attr": {// activity 属性，只针对 action_type=1 的情况
                "if": 0, // Intent 的 Flag 属性
                "pf": 0  // PendingIntent 的 Flag 属性
                 }
             }
         }
     }
}
```

### 客户端获取参数

1. 在您跳转指定的 activity 页面 onCreate 方法内，添加如下获取 intent uri 代码：

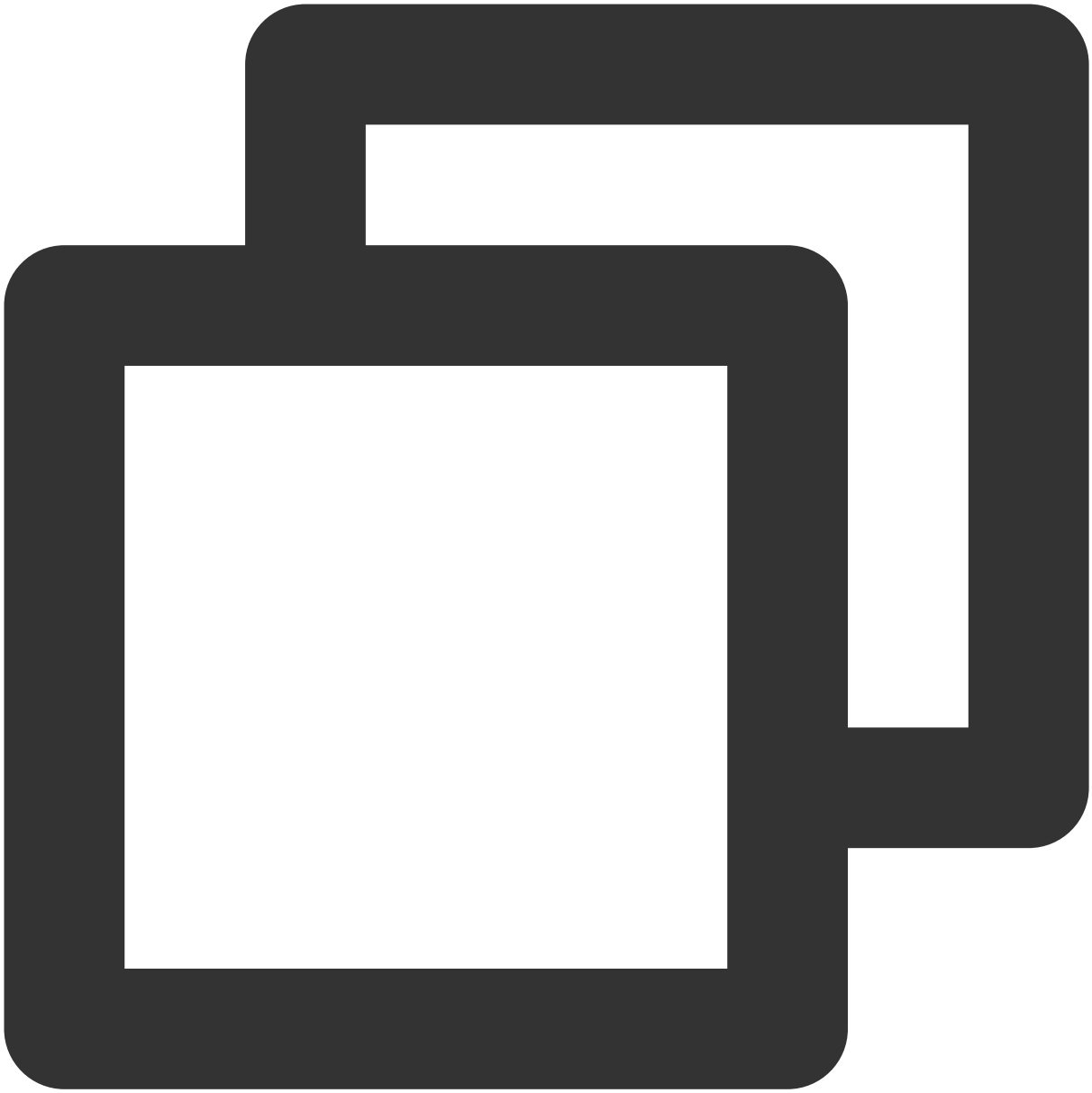


```
@Override
protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_temp);

 // ...

 // onCreate 内获取

 // 当使用 intent uri （客户端自定义）方式推送，且在 intent 尾部添加了参数，可以通过此段代码获取
 // 例如推送的 intent 值为 xgscheme://com.tpns.push/notify_detail?param1=aa&param2=bb
```

```
    // 则可以通过此段代码获取到 param1 的值 aa 和 param2 的值 bb。
    Uri uri = getIntent().getData();
    Log.i(TAG, "onCreate get data uri: " + uri);
    if (uri != null) {
        String url = uri.toString();
        String p1 = uri.getQueryParameter("param1");
        String p2 = uri.getQueryParameter("param2");
    }

    // SDK 1.3.2.0 起新增
    // 当创建推送任务时有填自定义参数（custom_content），可以通过此接口获取 custom_content 字符串
    String customContent = XGPushManager.getCustomContentFromIntent(this, this.getInte

    // ...
}
```

2. 如您的 activity 页面为应用常驻页面，例如配置了启动模式 launchMode 为 singleTop 或 singleTask，推送点击的 intent 内容将通过 activity 页面的 onNewIntent 方法触发，请同时在 onNewIntent 方法内添加如下获取 intent uri 代码。

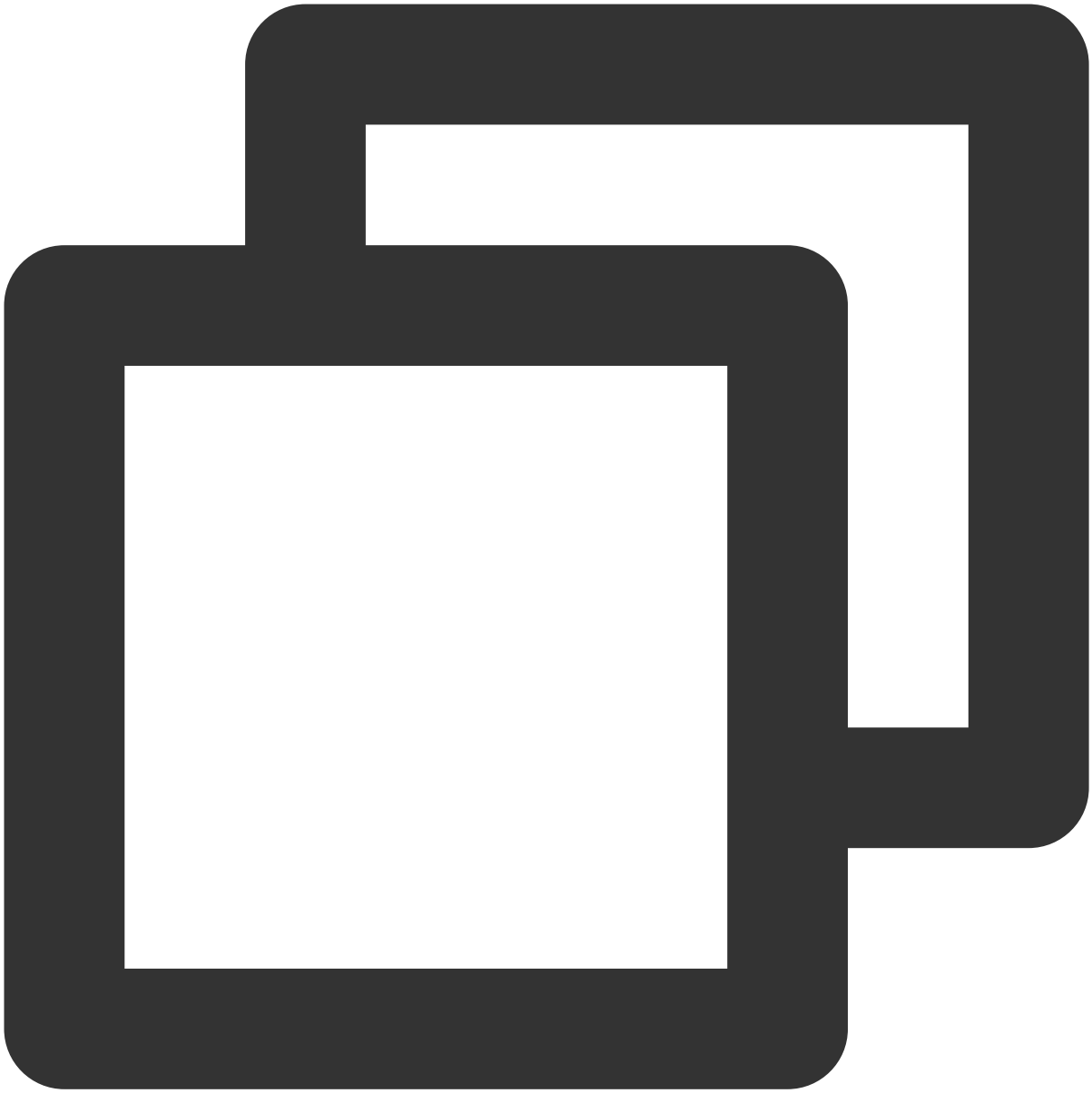
版权所有：腾讯云计算（北京）有限责任公司



```
@Override
protected void onNewIntent(Intent intent) {
 super.onNewIntent(intent);

 // ...

 // onNewIntent 内获取

 // 当使用 intent uri （客户端自定义）方式推送，且在 intent 尾部添加了参数，可以通过此段代码获
 // 例如推送的 intent 值为 xgscheme://com.tpns.push/notify_detail?param1=aa&param2=bb
 // 则可以通过此段代码获取到 param1 的值 aa 和 param2 的值 bb。
```

```
  Uri uri = intent.getData();
  Log.i(TAG, "onNewIntent get data uri: " + uri);
  if (uri != null) {
      String url = uri.toString();
      String p1 = uri.getQueryParameter("param1");
      String p2 = uri.getQueryParameter("param2");
  }

  // SDK 1.3.2.0 起新增
  // 当创建推送任务时有填自定义参数（custom_content），可以通过此接口获取 custom_content 字符串
  String customContent = XGPushManager.getCustomContentFromIntent(this, intent);

  // ...
 }
```

3.如果传参包含有特殊字符，可以在创建推送时将参数值进行 URLEncode，然后在终端内使用 URLDecode 进行解析，示例如下：



```
Uri uri = getIntent().getData();
if (uri != null) {
 String p1 = uri.getQueryParameter("param1");
 String value1 = "";
 try {
     // 自定义参数 param1 的值包含特殊字符，创建推送时可将 param1 的值进行 URLEncode；此处获取
     value1 = URLDecoder.decode(p1, "UTF-8");
 } catch (UnsupportedEncodingException e) {
     Log.w("TPNS", "URLDecode param failed: " + e.toString());
 }
 // 自定义参数 param2 未进行 URLEncode，直接获取即可
```

版权所有：腾讯云计算（北京）有限责任公司

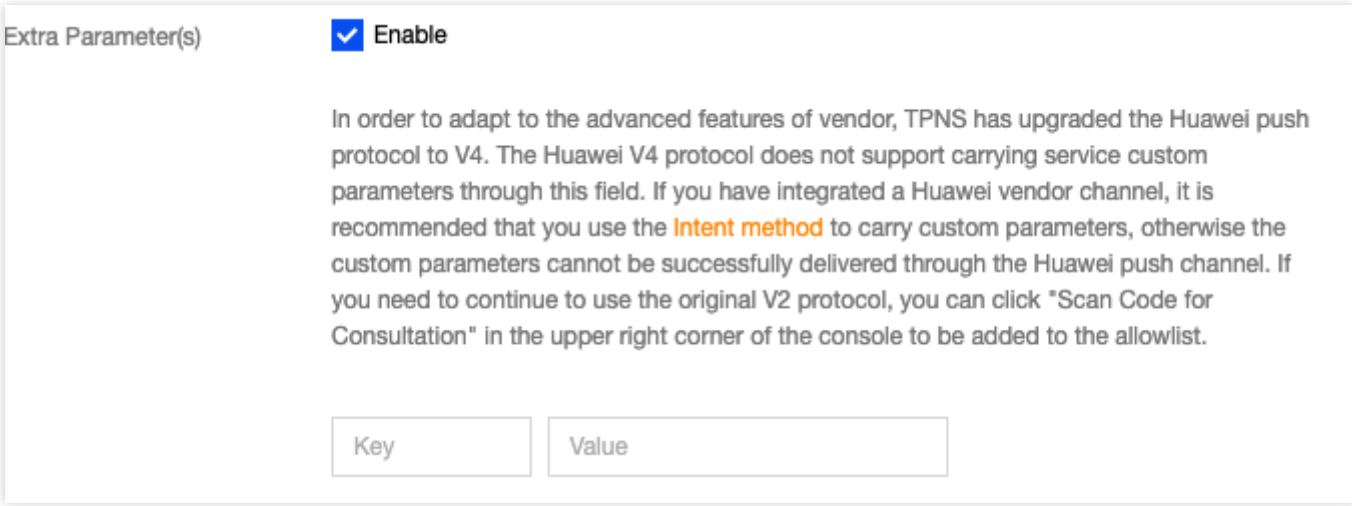
```
  String value2 = uri.getQueryParameter("param2");
  Log.i("TPNS" , "value1 = " + value1);
 }
```

# iOS 使用

用户可以在通知中携带自定义参数字段下发，通过解析客户端获取的参数自行实现跳转或者其他业务逻辑。

### 控制台使用

使用 移动推送控制台 设置参数，在**高级设置 > 附加参数**中填写如下：

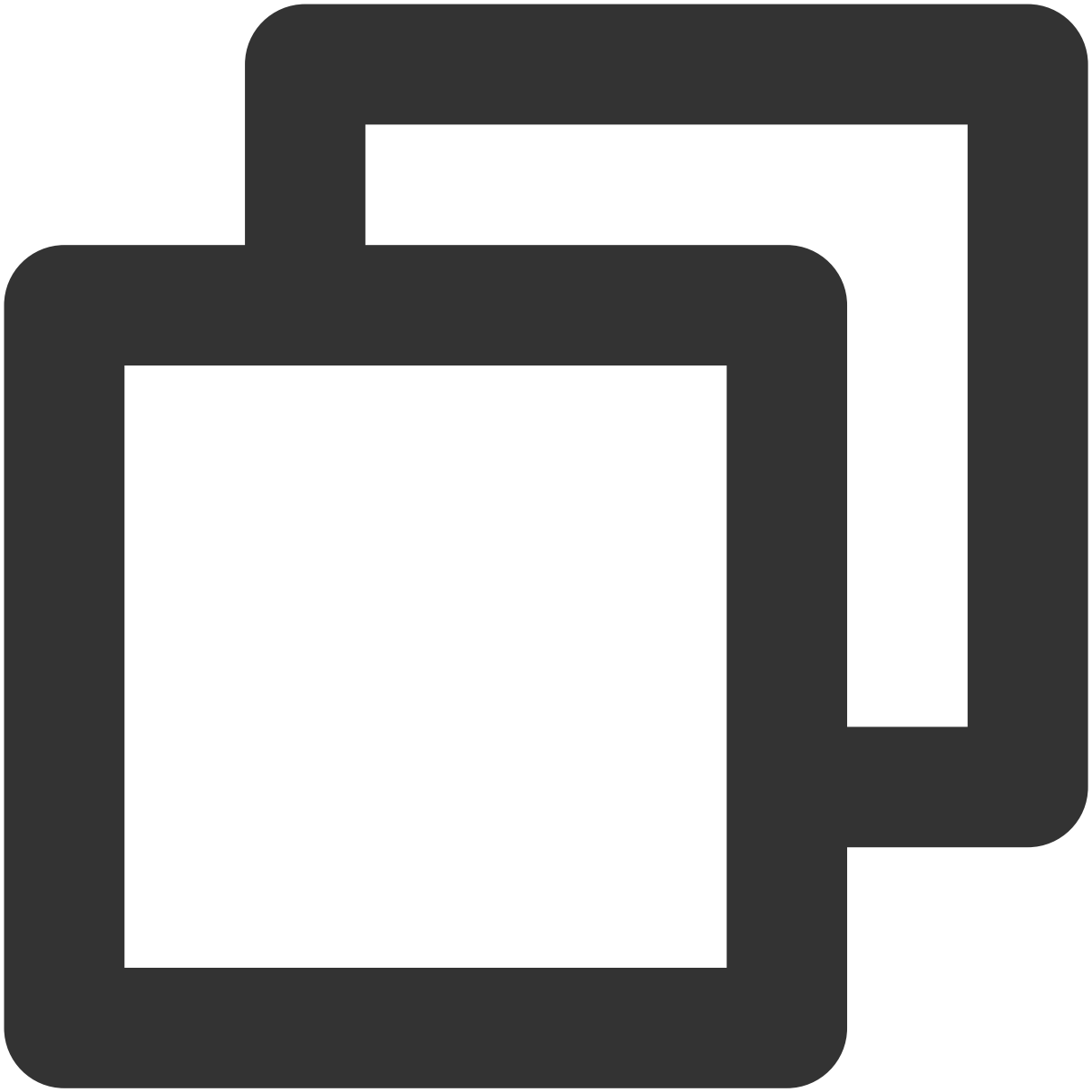



### Rest API 使用

在推送消息体中，`body.message.ios` 下添加 `custom_content` 字段，属性如下：

| 字段名 | 类型 | 父项目 | 默认值 | 必需 | 参数描述 |
| --- | --- | --- | --- | --- | --- |
| custom_content | String | ios | 无 | 否 | 自定义下发的参数，需要序列化为 json string |

完整的消息示例如下：

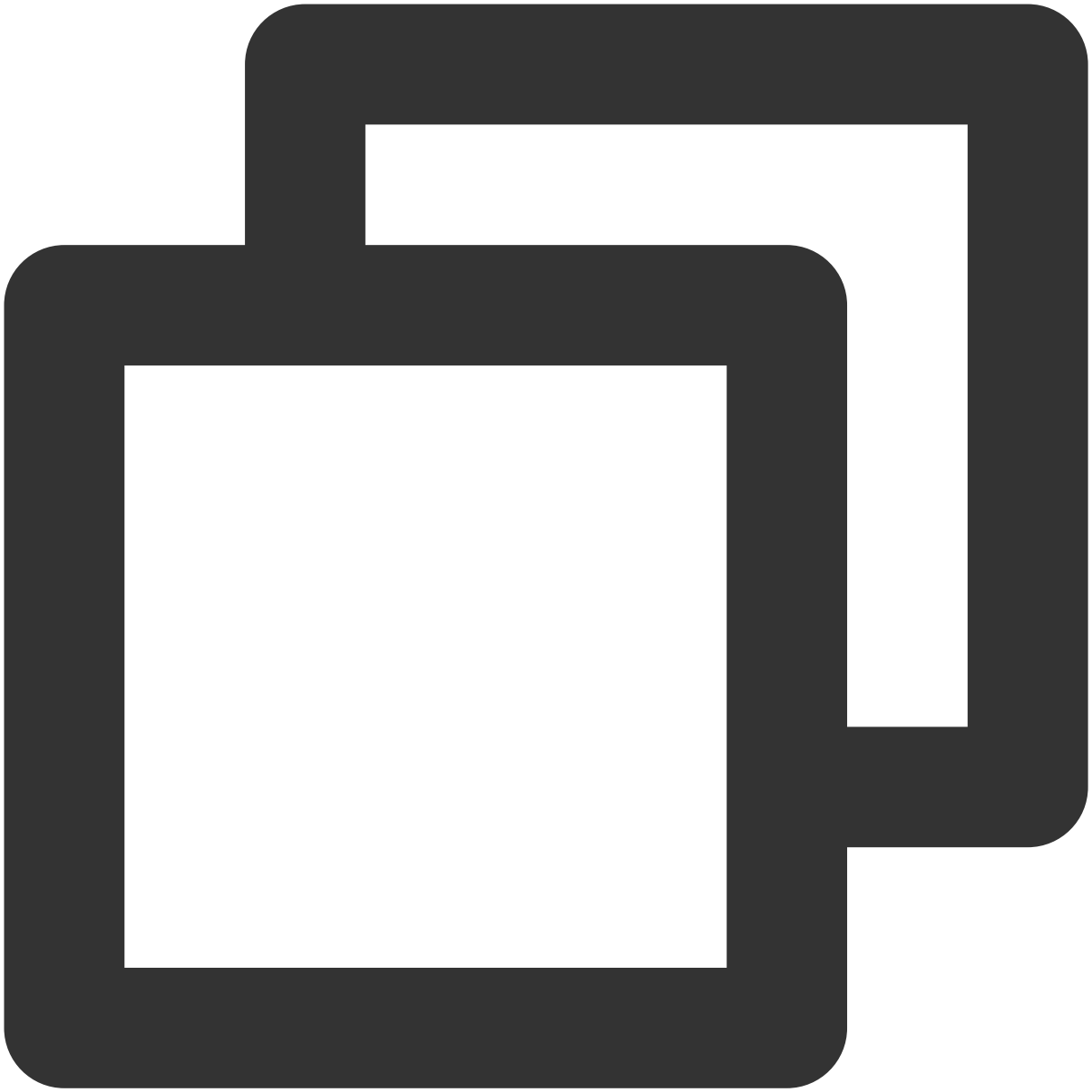


```
{
    "audience_type": "token",
    "environment": "dev",
    "token_list": [
        "0250df875c93c555dd3a2ba536b54fc1xxxx"
    ],
    "message_type": "notify",
    "message": {
        "title": "xxx",
        "content": "xxxxxxxxx",
        "ios": {
```

```
            "aps": {
                "alert": {
                    "subtitle": "xxx"
                }
            },
            "custom_content": "{\\"key\\":\\"value\\"}"
        }
    }
}
```

### 客户端获取参数

如果您通过 iOS SDK 集成，可以通过统一点击消息回调获取自定义参数，此回调方法是应用在所有状态（前台、后台、关闭）下的通知消息点击回调。



```
/// 统一点击回调
/// @param response 如果 iOS 10+/macOS 10.14+ 则为 UNNotificationResponse，低于目标版本
/// @note TPNS SDK1.2.7.1+
- (void)xgPushDidReceiveNotificationResponse:(nonnull id)response withCompletionHan
    NSLog(@"[TPNS Demo] click notification");
    if ([response isKindOfClass:[UNNotificationResponse class]]) {
        /// iOS10+消息体获取
        NSLog(@"notification dic: %@", ((UNNotificationResponse *)response).notific
    } else if ([response isKindOfClass:[NSDictionary class]]) {
        /// <IOS10消息体获取
        NSLog(@"notification dic: %@", response);
```

```
    }
    completionHandler();
}
```

版权所有：腾讯云计算（北京）有限责任公司

# 定速推送

最近更新时间：2024-01-16 17:34:39

定速推送是为了解决移动推送在推送消息时速度过快，导致部分客户服务器连接压力过大的问题，移动推送提供接口设置让用户按自身服务器情况控制推送速度。

## 场景

场景 1 ：
某 App 运营人员需要全量推送一条活动消息给用户，但活动页面对同时访问的人数有限制，希望能控制推送速度，缓减服务器连接压力，此时可设置定速推送，限制同时点开活动页面的人数。
场景 2 ：
某 App 对一批用户设置了"流失用户标签"，运营人员计划针对这批用户发送一条"福利领取"的推送消息，来吸引用户打开APP，同时不希望活动页面访问人数过多，此时可设置定速推送，限制同时点开道具领取页面的人数。

## 操作说明

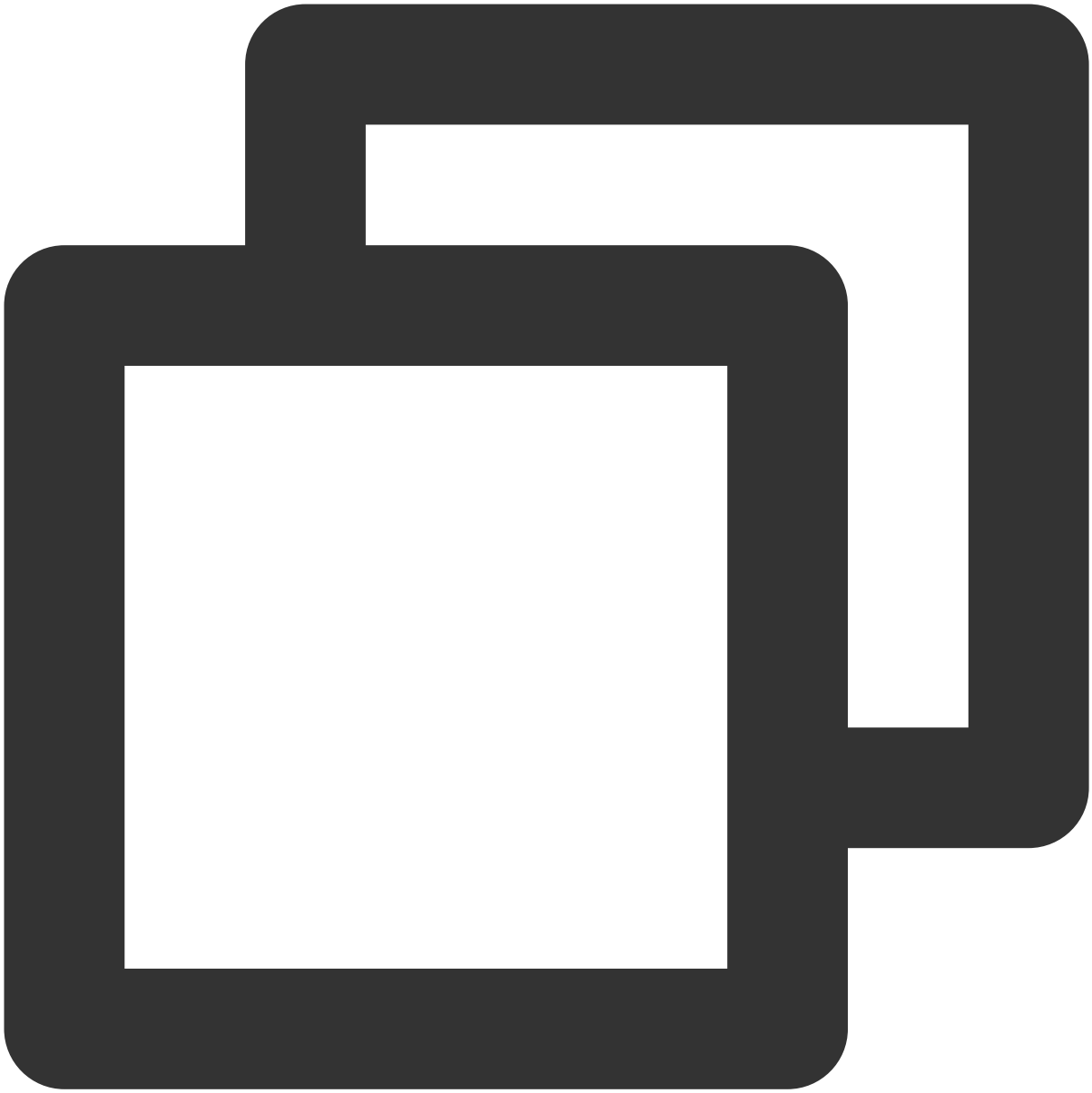
### 控制台使用

1. 登录 腾讯移动推送控制台。
2. 在左侧导航栏中，选择**推送管理** > **推送任务**，进入推送任务管理页面。
3. 单击**新建推送**，并在**高级设置**的“定速推送”栏，勾选**开启**，选择推送速度。

开启定速推送后，推送将按照选择的速度对匹配推送目标的设备进行下发。
**说明：**
仅全量推送和标签推送支持定速推送功能。
推送速度有效值范围1000~50000条/秒。

### Rest API 使用

在 Rest API 可选参数中设置 `push_speed` 参数，可实现定速推送，具体参考 PushAPI 参数说明。
推送示例如下：



```
{
   "audience_type": "tag",
    "tag_list": {
        "tags": [
            "tag1",
            "tag2"
        ],
        "op": "AND"
    },
    "push_speed":50000,
    "message_type": "notify",
```

```
    "message": {
        "title": "推送标题",
        "content": "推送内容",
        "android": {
            "custom_content":"{\\"key\\":\\"value\\"}"
        }
    }
}
```

版权所有：腾讯云计算（北京）有限责任公司

# 通道策略

最近更新时间：2024-01-16 17:34:39

随着厂商通道推送额度和推送频率的限制逐步收紧，push 推送的抵达率和下发速度也受到对应程度的限制。具体限制可参见：

厂商通道限额说明

厂商通道 QPS 限制说明

移动推送提供「智能分配」和「自定义」两种通道分配策略，可在厂商通道限制下提升推送的综合抵达率和抵达速度。

## 通道简介

| 通道类型 | 适用条件 | 支持的手机品牌 |
| --- | --- | --- |
| 移动推送自建通道 | App 进程在线 | 所有品牌手机 |
| Android 厂商推送通道（华为、小米、魅族、vivo、OPPO、FCM） | App 进程在线或离线 | 华为、小米、OPPO、vivo、魅族、一加、黑鲨、realme、iQOO、荣耀、国外具备谷歌服务框架的手机 |
| iOS 厂商推送通道（APNs） | App 进程在线或离线 | Apple |

## 通道策略概览

### 智能分配

移动推送会结合设备状态、人群活跃状态和推送通道状态，智能为每个设备分配最佳的下发通道，以此达到以下效果：

1. 提升推送综合抵达率。
2. 提高推送综合抵达速度。
3. 节省部分厂商通道可用额度。

### 自定义通道策略

目前各厂商通道限制每日推送额度，您可根据业务需求，选择某条推送任务可以通过哪些通道下发，个性化地调整 push 通道下发策略，以节省厂商通道资源，实现推送的价值最大化。

自定义策略详细下发规则见下表：

| | | | |
| --- | --- | --- | --- |

| 通道 | 开启 | 关闭 | 支持的消息类型 |
| --- | --- | --- | --- |
| Android 厂商推送通道（华为、小米、魅族、vivo、OPPO、FCM） | 本次推送可用通道为厂商推送通道、移动推送自建通道。<br>注意：<br>若开启**在线优先 移动推送**，则设备在线时优先通过移动推送自建通道下发。<br>若关闭**在线优先 移动推送**，则优先通过厂商通道下发。<br>当厂商通道推送失败时，会通过移动推送自建通道补推。<br>当关闭自建通道时，只能通过厂商通道下发。 | 关闭两个通道的其中一个通道，自建通道或者厂商通道，不能同时关闭 | 通知栏消息 |
| iOS 厂商推送通道（APNs） | 本次推送可用通道为 APNs 推送通道、移动推送自建通道。<br>注意：<br>若开启**在线优先 移动推送**，则设备在线时优先通过移动推送自建通道下发。<br>若关闭**在线优先 移动推送**，则优先通过 APNs 通道下发。<br>当 APNs 通道推送失败时，会通过移动推送自建通道补推。<br>当关闭自建通道时，只能通过厂商通道下发。 | 关闭两个通道的其中一个通道，自建通道或者厂商通道，不能同时关闭 | 通知栏消息和静默消息<br>**注意**：APNs 通道静默消息限频（单个设备3条/小时） |
| 移动推送 | 本次推送可用通道为移动推送 | 可关闭 | 通知栏消息、透传消息、静默消息<br>**注意**：iOS 端移动推送自建通道仅在 iOS SDK 1.2.8.0及以上版本生效 |

# 开始使用

### 控制台使用

您可在控制台创建推送时选择该条推送的通道策略，具体路径如下：

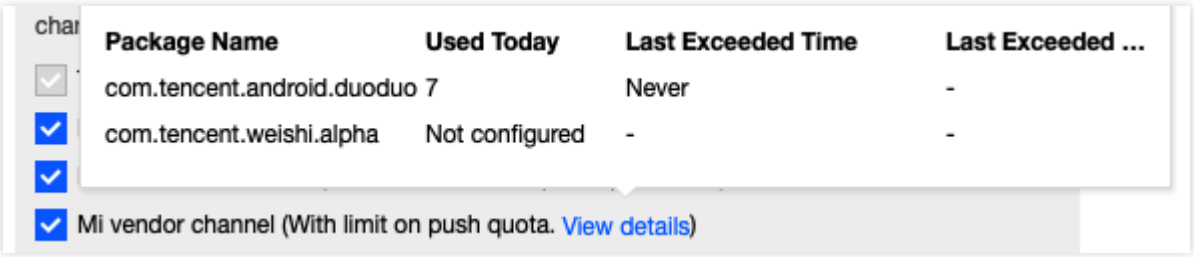
控制台 > **App推送管理 > 推送任务 > 新建推送 > 高级设置 > 通道策略**

#### 智能分配

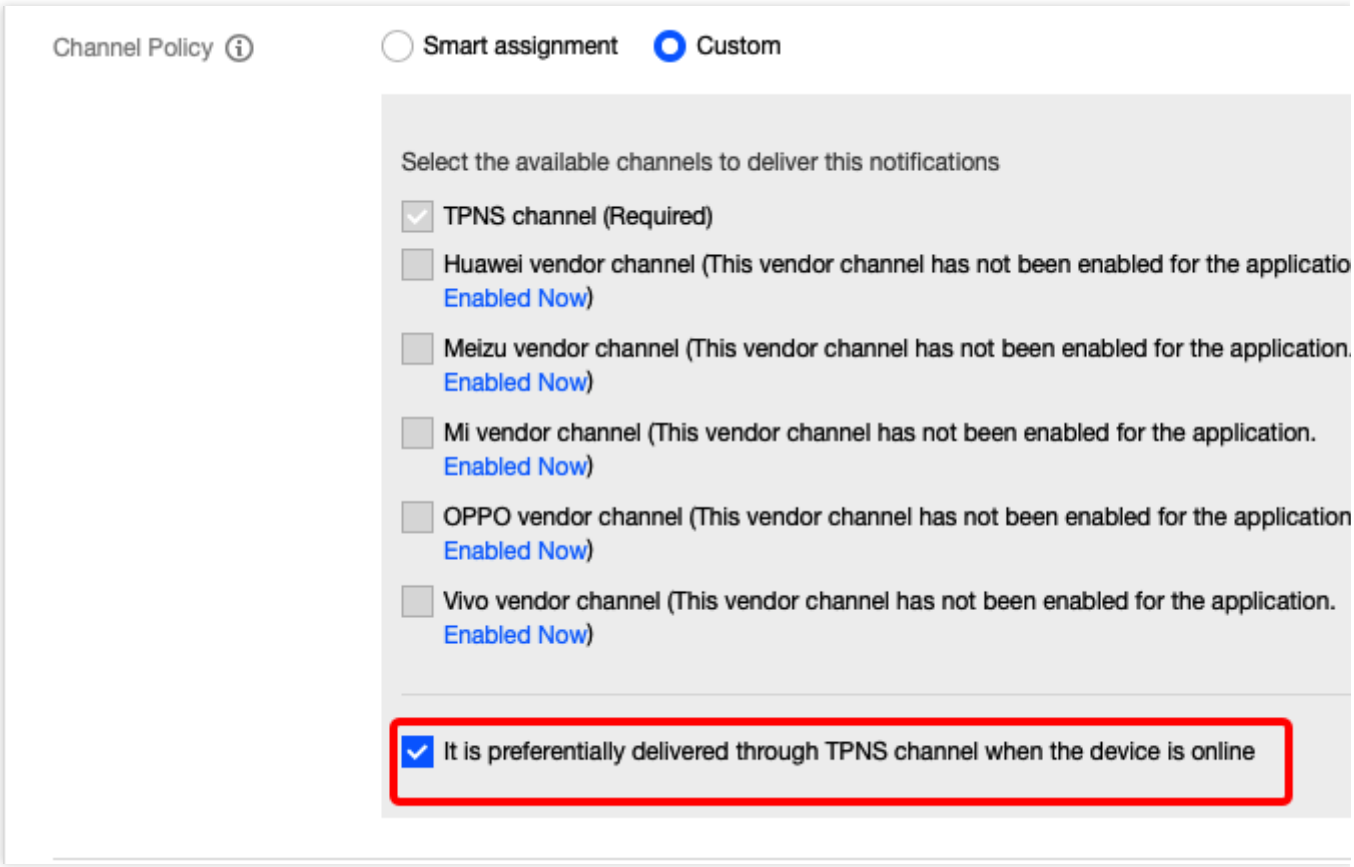
选择**智能分配**，系统会智能分配每个设备的下发通道，详情请参见 智能分配 的规则。

### 自定义 Android 通道策略

选择**自定义**，单击**查看详情**可以查看详细的厂商额度信息。




可以根据当前厂商通道剩余配额，以及推送任务的优先级，自定义选择需要推送的通道，详情请参见 自定义通道策略 的规则，为了节省各厂商通道推送资源，建议开启**设备在线时优先通过移动推送自建通道下发**。

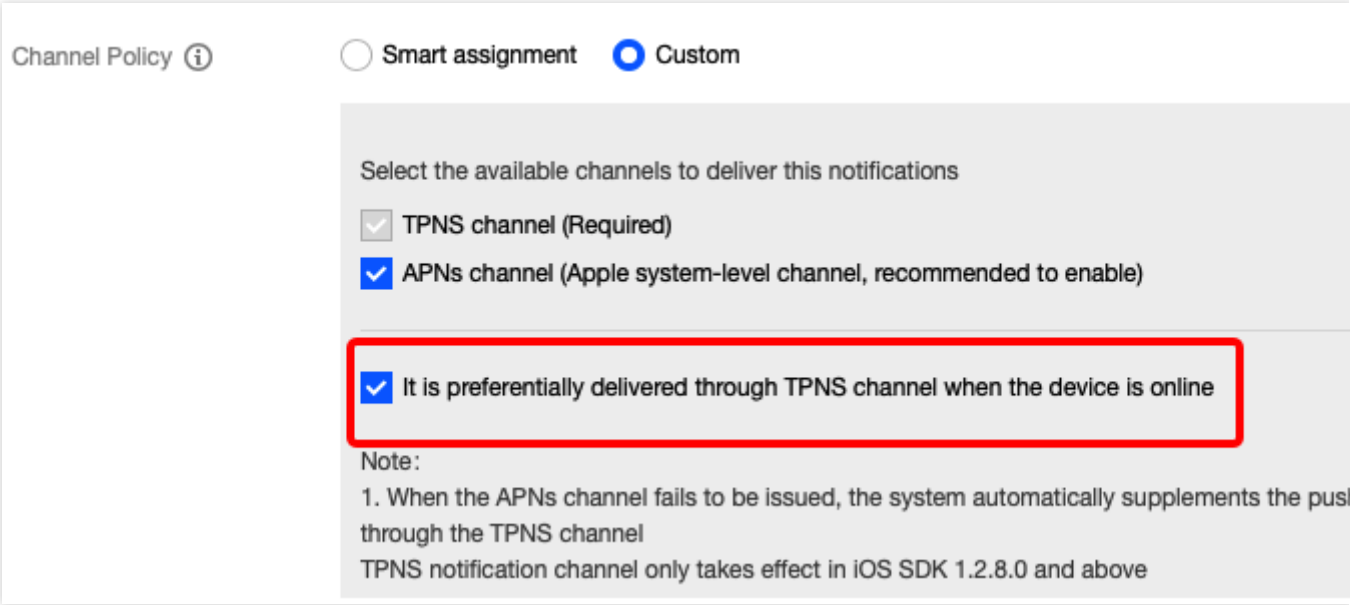



**注意：**

移动推送自建通道可关闭。

### 自定义 iOS 通道策略

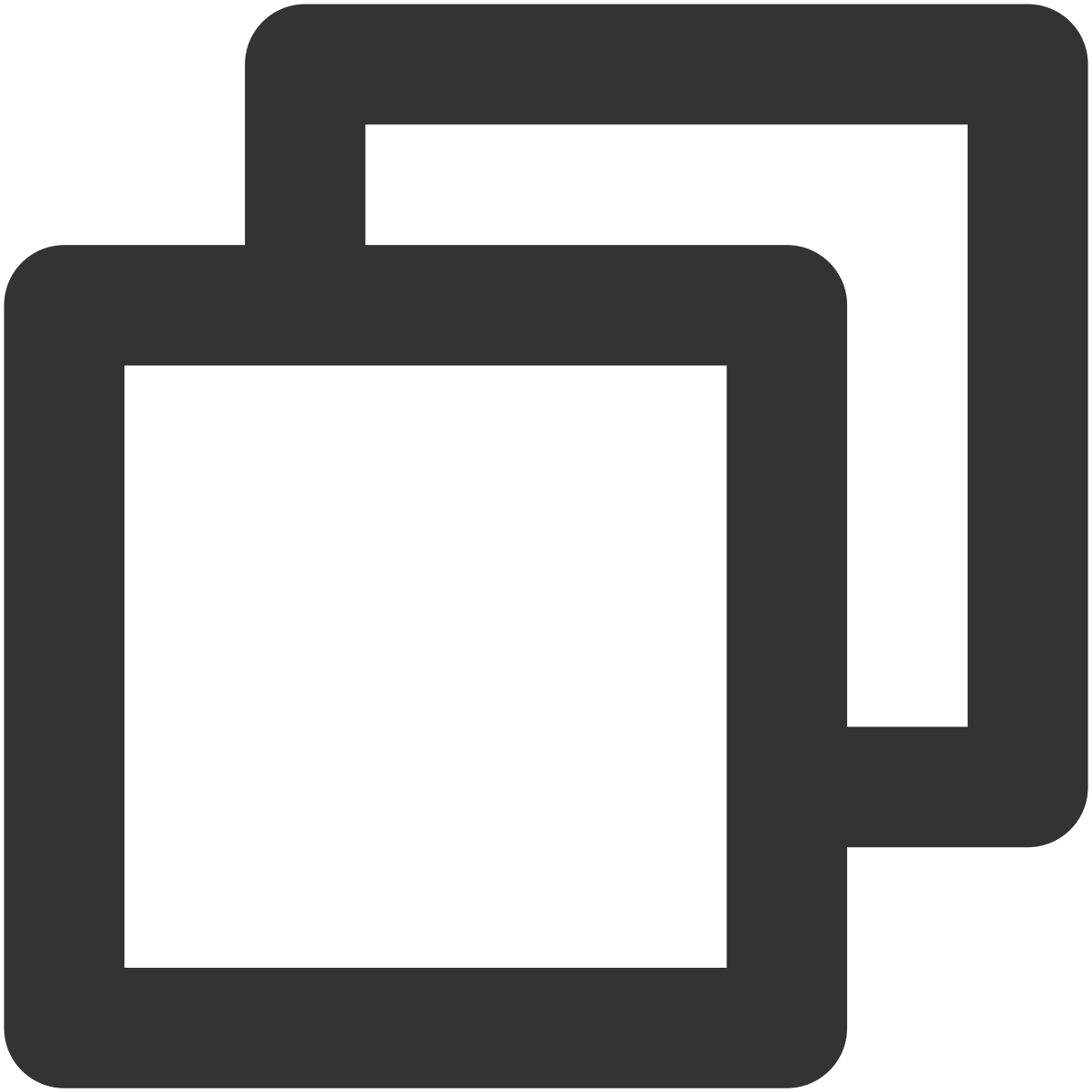
可根据您推送任务的优先级，自定义选择需要推送的通道，详情请参见 自定义通道策略 的规则，建议开启【设备在线时优先通过移动推送自建通道下发】以保证通知最快速的抵达设备。




## Rest API 使用

在 Rest API 可选参数中设置通道策略（channel_rules）参数，可自由设置通道策略，详情请参见 PushAPI 文档中的 channel_rules 参数说明。

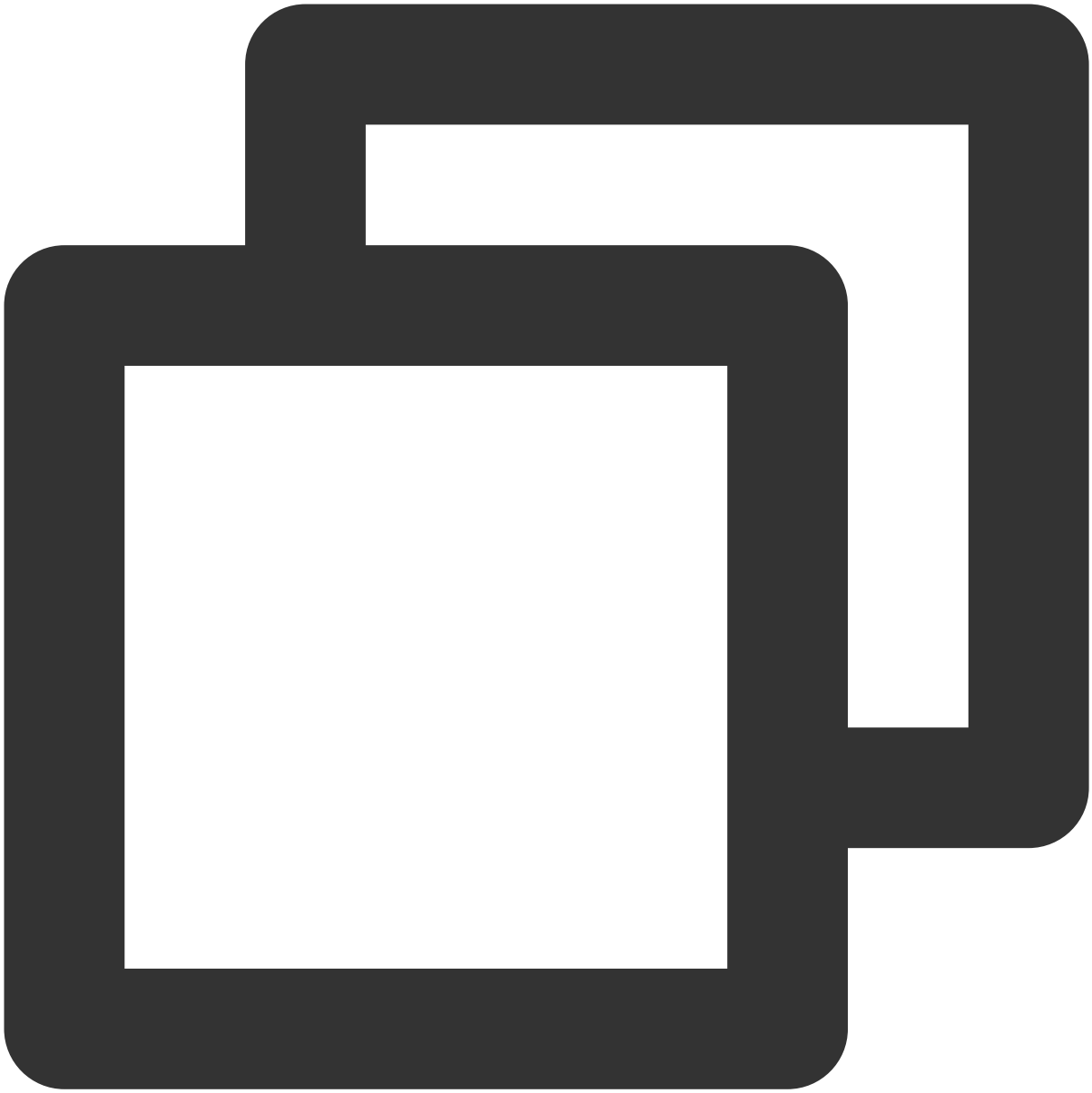
Android 端推送示例如下：



```
{
    "audience_type": "token",
    "token_list": [
        "05da87c0ae*************8d884aada5bb2"
    ],
    "message_type": "notify",
    "channel_rules": [
        {
            "channel": "mz",
            "disable": true   //关闭魅族通道
        },
```

```
        {
            "channel": "xm",
            "disable": false  //打开小米通道
        }
    ],
    "tpns_online_push_type":0, //默认在线走移动推送自建通道下发
    "message": {
        "title": "此推送通过小米通道下发且不需要通过魅族通道下发",
        "content": "推送内容",
        "android": {
             "custom_content":"{\\"key\\":\\"value\\"}"
        }
    }
}
```

iOS 端推送示例如下：



```
{
        "audience_type": "token",
        "environment": "dev",
        "token_list": ["05da87c0ae********fa9e08d884aada5bb2"],
        "message_type": "notify",
        "channel_rules": [{
            "channel": "apns",
            "disable": true
        }],
        "tpns_online_push_type": 0,
        "message": {
```

```
            "title": "本次推送只通过移动推送自建通道下发",
            "content": "推送内容",
            "ios": {
                "aps": {
                    "alert": {
                        "subtitle": "推送副标题"
                    },
                    "badge_type": -2,
                    "sound": "Tassel.wav"
                },
                "custom_content": "{\\"key\\":\\"value\\"}"
            }
        }
}
```

# 消息折叠

最近更新时间：2024-01-16 17:34:39

## 操作场景

在手机通知中心，当收到来自同一个 App 的多条通知时，系统会根据对应规则，将多条通知折叠起来，以免打扰用户。这样会减少 push 的曝光率，继而降低 push 运营的收益。
Android 原生系统和 iOS 系统都提供了相应设置方式，开发者可以根据业务的运营需求，通过此功能设置通知折叠分类，将相同类型的通知折叠在一起，提高通知的可读性。

## 适用范围

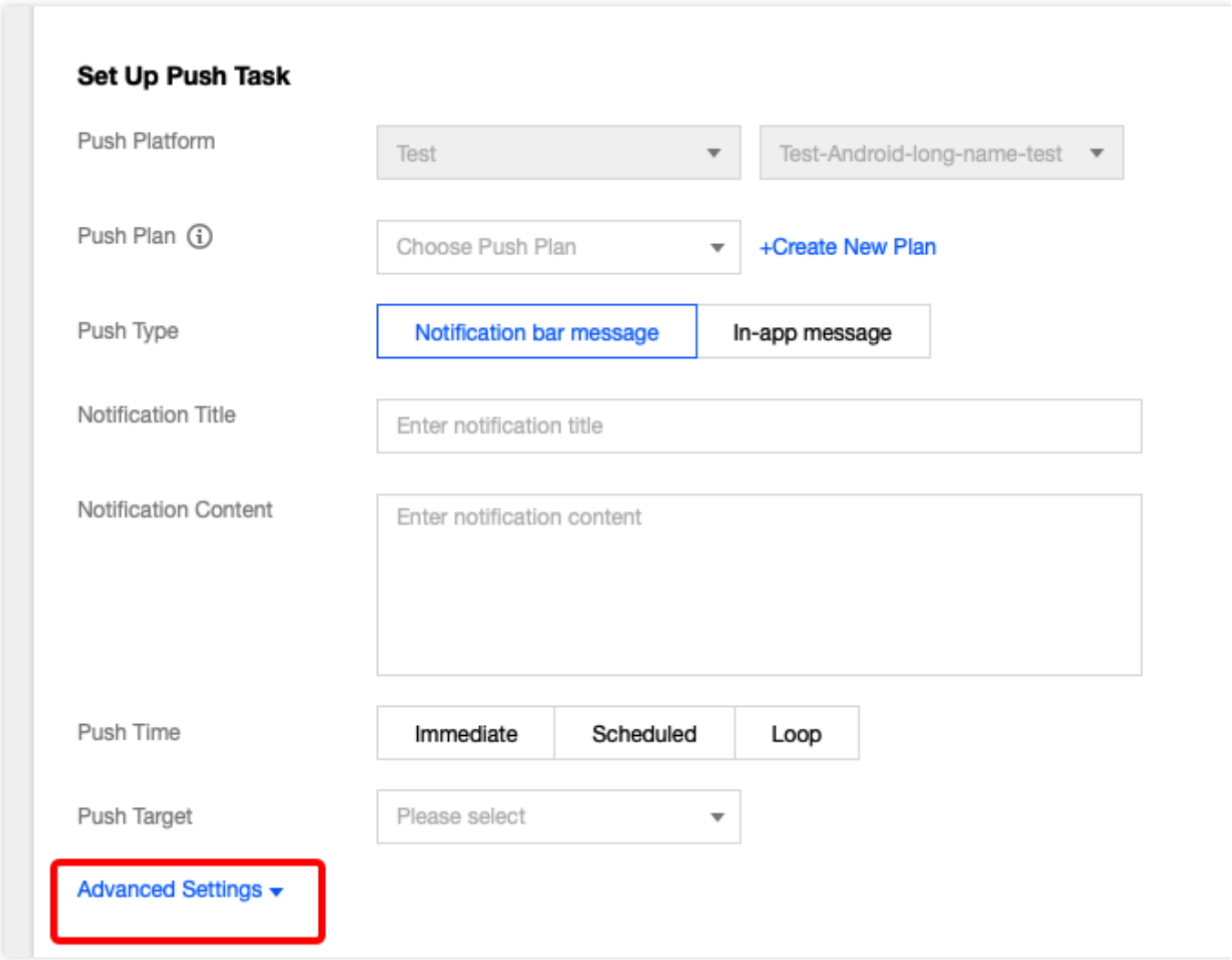
| 手机系统 | 系统版本 | SDK 版本 | 推送通道 |
| --- | --- | --- | --- |
| Android | Android 7.0 + | TPNS SDK 1.2.0.1 + | 移动推送自建通道 |
| iOS | iOS 10 + | TPNS SDK 1.2.0.1 + | APNs 通道 |

## 操作步骤

### 控制台使用

1. 登录 移动推送控制台，在左侧导航中选择**推送管理** > **推送任务**，单击**新建推送** > **高级设置**。

版权所有：腾讯云计算（北京）有限责任公司




2. 设置消息折叠规则，如下图所示：

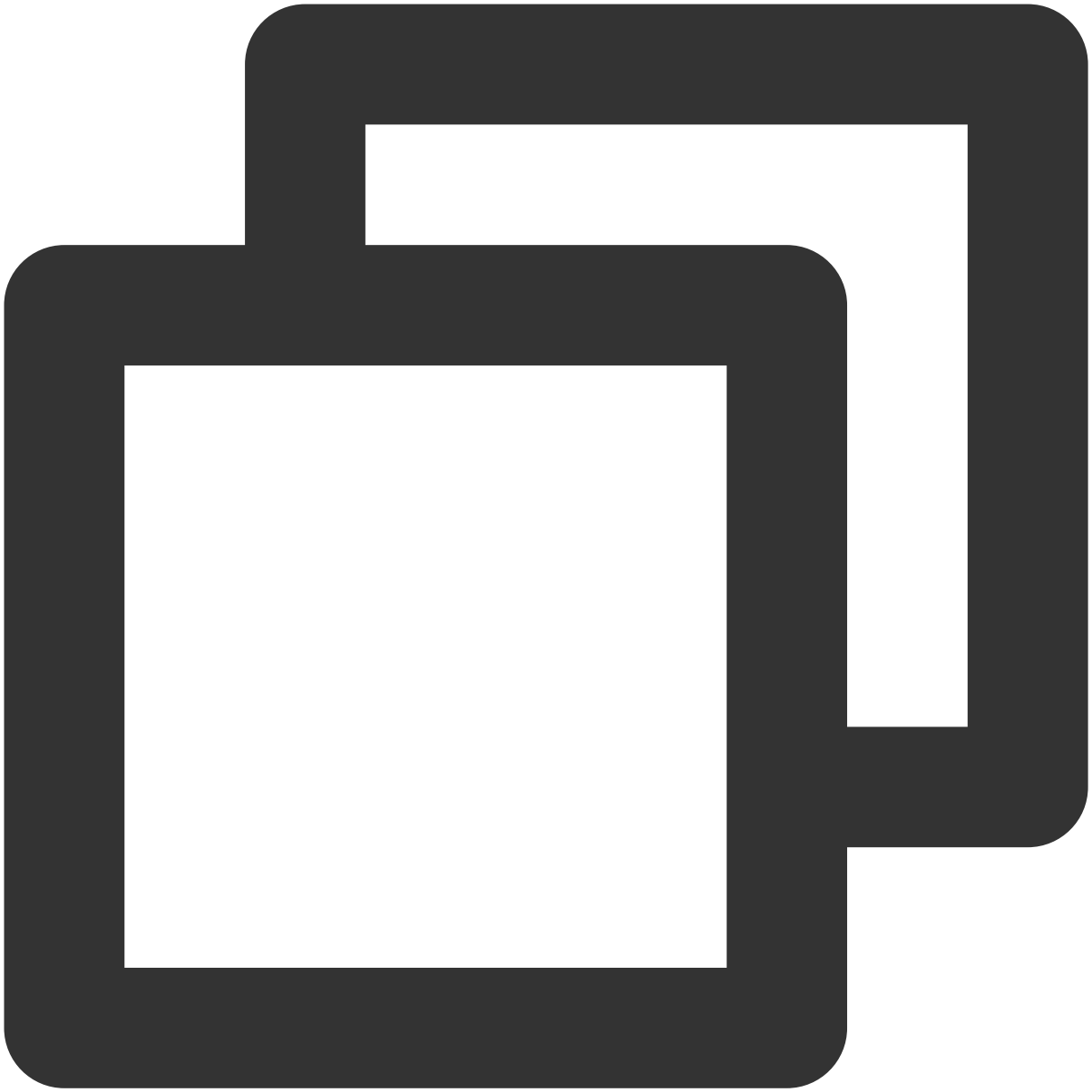
控制通知在设备通知中心是否折叠以及折叠方式有以下三种选项：
系统默认：跟随操作系统的默认折叠规则
部分厂商系统规则如下：
原生安卓：同一应用通知达到5条时折叠成一组。折叠时右侧数字为被折叠的未阅读通知总量，展开时最多显示8条通知内容。
华为：同组通知达到2条时开始分组。折叠时右侧数字为被折叠的未阅读通知总量，展开时最多显示8条通知内容。
小米：同组通知达到4条时开始分组，折叠时右侧数字为被折叠的通知数量（最大为7），展开时最多显示10条通知内容。
魅族：同组通知达到4条时开始分组，折叠时顶部数字为通知总量，展开时可展示的通知条数很多，目前已知超过35条。
**说明：**

魅族手机提供的不重要通知功能，会将过多的通知收纳到右上角收纳盒子中，这些通知不会被折叠。可以通过在应用通知设置内开启“设为优先”来关闭此功能。
OPPO：同组通知达到 4条时开始分组，折叠时右侧数字为被折叠的未阅读通知总量，展开时最多显示8条通知内容。
vivo：同组通知达到 2条时开始分组，折叠时无数字显示被折叠的通知总量，展开时最多显示8条通知内容。
不折叠：通知不会与相同应用其他通知折叠 。
自定义：携带相同 `thread_id` 的通知会被合并折叠。
**说明：**
若通知优先级不高可通过设置此参数，将此类通知折叠，避免过多打扰用户，若通知比较重要希望用户优先看到，可设置为不折叠。

## Rest API 使用

若想要实现控制台“不折叠”和“自定义”的效果，则需要在 Push API `message` 字段中自定义设置 `thread_id` ，详情可参考 PushAPI 可选参数 说明。
推送示例如下：

版权所有：腾讯云计算（北京）有限责任公司



```
{
    "audience_type": "token",
    "token_list": [
        "05da87c0ae******2dfa9e08d884aada5bb2"
    ],
    "message_type": "notify",
    "multi_pkg":true,
    "message": {
        "title": "推送标题",
        "content": "推送内容",
        "android": {
```

```
                "custom_content":"{\\"key\\":\\"value\\"}"
            },
            "thread_id":"活动_id",
            "thread_sumtext":"运营活动"

        }
    }
```

# 富媒体通知

最近更新时间：2024-01-16 17:34:39

## 操作场景

富媒体推送允许开发者推送图片、声音、视频等除普通文本之外更丰富的内容，能够有效提升通知点击率。应用开发者可以利用该功能推送如新闻、优惠券、活动信息等更加丰富的内容，满足不同开发者的个性化推送需求。

## 适用范围

目前 移动推送 富媒体推送功能支持的富媒体程度如下：

Android：支持图片、音频。

iOS：支持图片、音频、视频。

**说明：**

iOS 平台 移动推送自建通道默认支持富媒体推送功能，APNs 通道需要 App 集成 通知服务扩展插件。

各推送通道富媒体支持类型和使用要求如下表所示：

| 推送通道 | 支持类型 | 使用要求 |
| --- | --- | --- |
| 华为 | 缩略图 | URL 仅支持 HTTPS<br>格式要求：<br>PNG/JPG/JPEG 格式<br>尺寸120px × 120px，若图片长或宽超过120px，华为系统会自动缩放<br>小于200KB |
| 小米 | 大图 | URL 仅支持 HTTPS<br>格式要求：<br>PNG/JPG/JPEG 格式<br>固定876px x 324px<br>小于1MB<br>**注意**：如需使用小米通道大图通知功能，需先调用小米图片上传接口上传图片文件，获取小米指定的图片地址 pic_url，再填入移动推送 推送对应的参数 xg_media_resources 中。详情请参见小米推送富文本消息文档中的上传大图 API 部分 |
| 移动推送 | 大图+缩略图<br>+音频 | Android 平台：<br>URL 仅支持 HTTPS<br>大图、缩略图格式要求：<br>JPEG/JPG/PNG 格式<br>大图高度不能超过324px，宽度自适应 |

| | | |
|---|---|---|
| | | 缩略图尺寸120px × 120px，若图片长或宽超过120px，移动推送会居中截取方形<br>音视频文件格式要求：音频文件大小不超过5M<br>iOS 平台：使用要求同 APNs 通道 |
| APNs | 缩略图+音视频 | URL 仅支持 HTTPS<br>图片格式要求：<br>JPEG/PNG/GIF 格式<br>小于10MB<br>音视频文件格式要求：<br>视频文件编码格式 MPEG/MPEG2video/MPEG4/AVI 格式<br>音频文件后缀名需要为 aiff，wav、caf 的一种<br>音频编码格式需要为 Linear PCM、MA4（IMA/ADPCM）、alaw、μLaw 的一种<br>音视频文件小于5MB<br>音频时长要求30s以下<br>说明：图片不能与通知音视频同时开启，只能开启其中一个 |

# 操作步骤

### 控制台使用

1. 登录 移动推送控制台，在左侧导航中选择**推送管理** > **推送任务**，进入推送任务页面。
2. 单击**新建推送** > **高级设置**。

3. 开启**通知图片**或**通知音视频**，填入富媒体消息 URL，详细配置说明如下：

当您开启**通知图片**时：

Android 端：

**说明：**

缩略图：

URL 格式必须为 HTTPS

通过移动推送自建通道和华为通道下发的推送可展示缩略图。

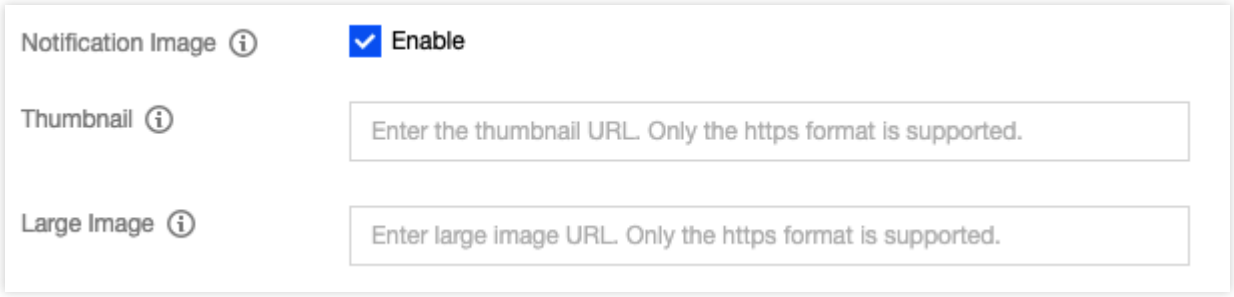
格式要求：`PNG/JPG/JPEG` 格式图片，尺寸120 × 120px，小于200KB。

大图：

URL 格式必须为 HTTPS

通过移动推送自建通道和小米通道下发的推送可展示大图。

格式要求：`PNG/JPG/JPEG` 格式图片，尺寸固定876 x 324px，小于1M。




iOS 端：

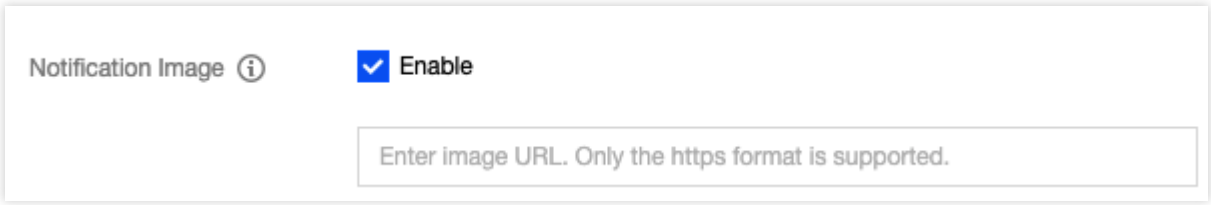
**说明：**

填写图片 URL 后，通知将展示图片，格式要求：

大小不超过10M

格式为 `PNG/JPG/JPEG/GIF`

URL 格式必须为 HTTPS




当您开启**通知音频**或**通知音视频**时：

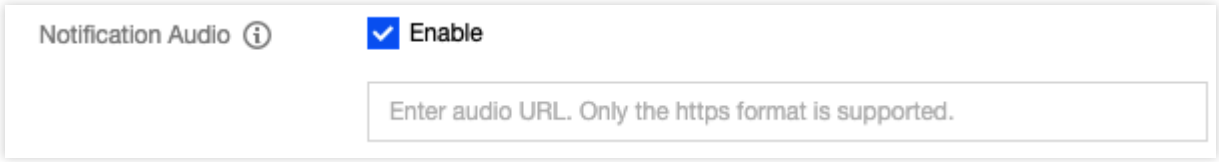
Android 端：

**说明：**

填写音频 URL 后，移动推送自建通道下发的通知可携带音频。

音频文件大小不超过5M

URL 格式必须为 HTTPS




iOS 端：

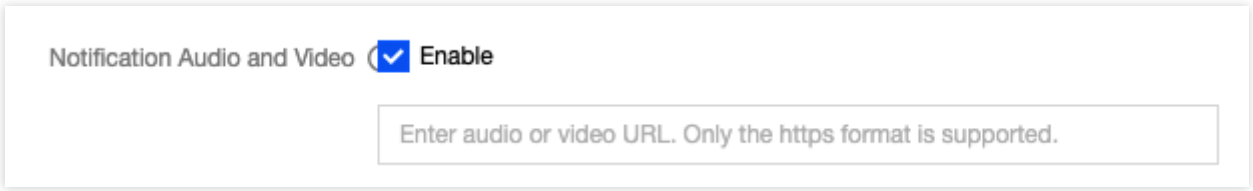
**说明：**

填写音视频 URL 后，通知可携带音频或视频，并通过 Apple 原生组件播放。

音视频文件大小不超过5M

格式为 `MPEG/MPEG2video/MPEG4/AVI`

URL 格式必须为 HTTPS



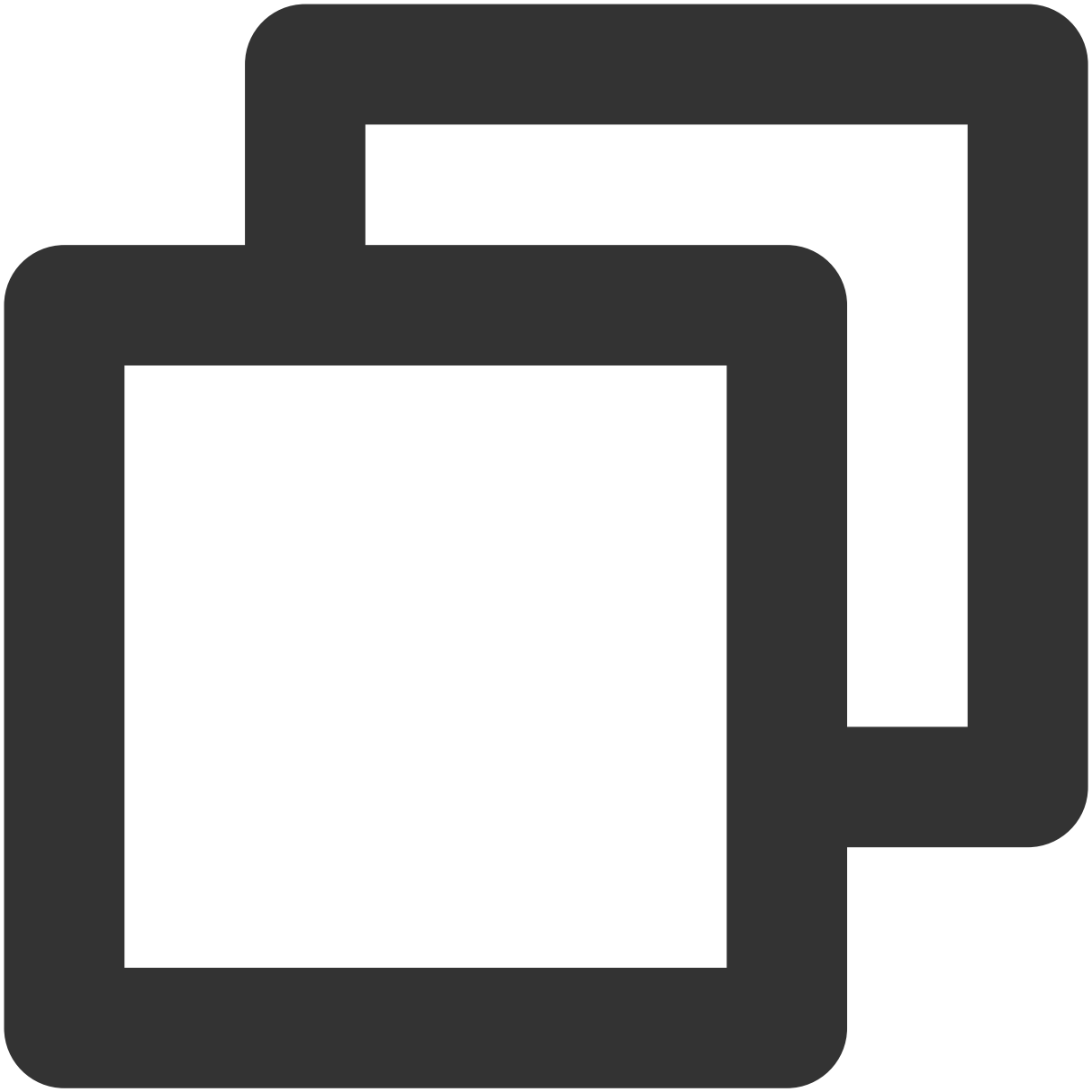

## REST API 使用

若需要在调用 API 推送时下发富媒体消息，可通过 Push API 中 Android 或 iOS 消息体中以下参数设置：

| 平台 | 参数 | 备注 |
| --- | --- | --- |
| Android | 缩略图：icon_res、icon_type<br>大图：xg_media_resources<br>音频：xg_media_audio_resources | 缩略图仅移动推送自建通道和华为通道支持<br>大图仅 移动推送自建通道和小米通道支持<br>音频仅移动推送自建通道支持 |
| iOS | 图片、音视频：xg_media_resources | 通知图片不能与通知音视频同时设置，只能设置其中一个 |

详情可参见 Push API 必要参数 说明。

**Android 端推送示例如下：**



```
{
    "audience_type": "token",
    "token_list": [
        "05da87c0ae******2dfa9e08d884aada5bb2"
    ],
    "message_type": "notify",
    "multi_pkg":true,
    "message": {
        "title": "推送标题",
        "content": "推送内容",
        "xg_media_resources": "xxx1" , //此处填富媒体元素地址，例如 https://www.xx.com/
```

```
          "xg_media_audio_resources":"xxx", //此处填音频富媒体元素地址，例如 https://sc1.1
          "android": {
         "icon_type": 1,
          "icon_res": "xxx", //此处填图片缩略图元素地址
               "custom_content":"{\\"key\\":\\"value\\"}"
          }
     }
}
```

# 常见问题

**魅族、OPPO、vivo 厂商通道不支持富媒体，推送全部设备时，通知如何下发？**

移动推送、华为、小米、FCM 通道下发的通知带图片，魅族、OPPO、vivo 通道下发的通知默认纯文本样式，不带图片。

**推送消息需要携带图片怎么使用？**

无论是调用 API 还是通过控制台下发，都需要将图片生成 URL 后使用。

**Android 音频富媒体下发策略是怎么样的？**

移动推送自建通道下发的音频富媒体推送，正常展示音频，其他通道下发的音频富媒体推送，默认纯文本样式，不带音频。

**iOS 富媒体通知图片和通知音视频可以同时开启吗？**

不能同时开启，只能开启其中一个。

版权所有：腾讯云计算（北京）有限责任公司

# 个性化通知

最近更新时间：2024-01-16 17:34:39

## 简介

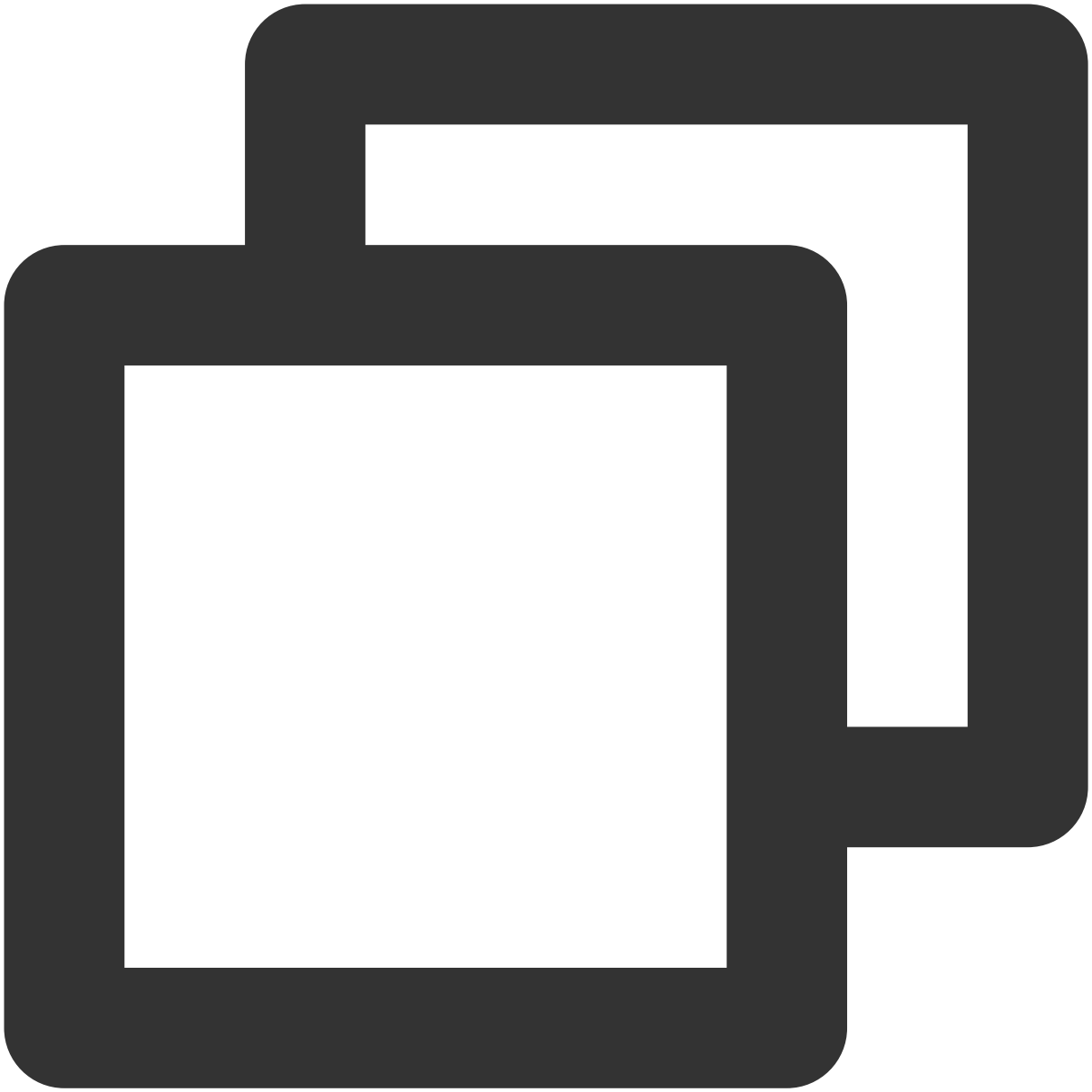
移动推送支持在推送文案中带上个性化参数，用户将个性化参数与设备绑定，创建推送时，终端将展示带有个性化参数的推送；相较于千篇一律的统一推送文案，用户点击意愿会更加强烈。

## 常见场景

### 电商应用场景

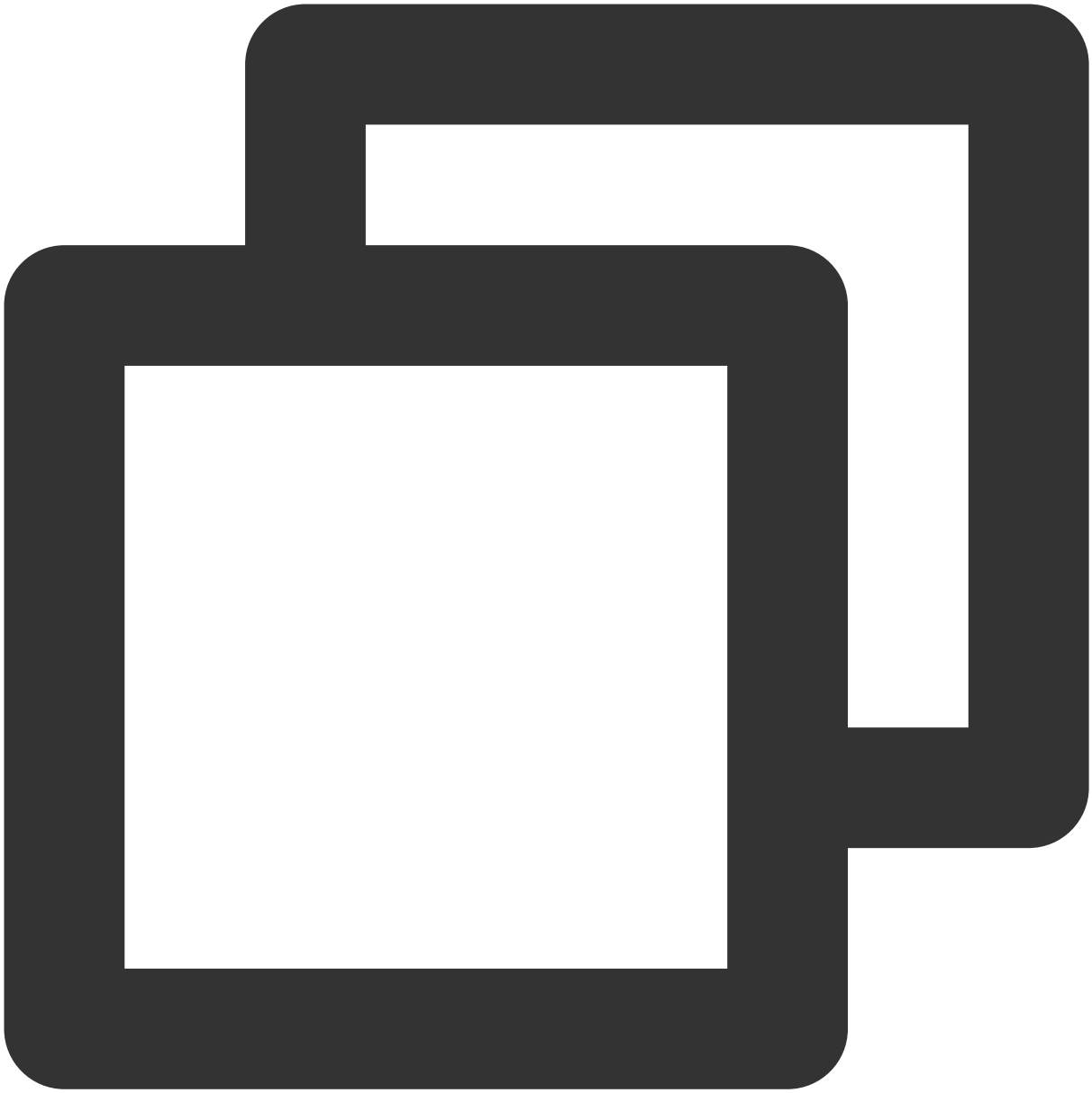
要提升用户购物车内商品的付费率，可使用以下模板：

版权所有：腾讯云计算（北京）有限责任公司



```
Hi @{{nickname}}, 你的购物车里还有一件{{productname}}, 库存只有{{productnum}}件！
快把它买了吧？
```

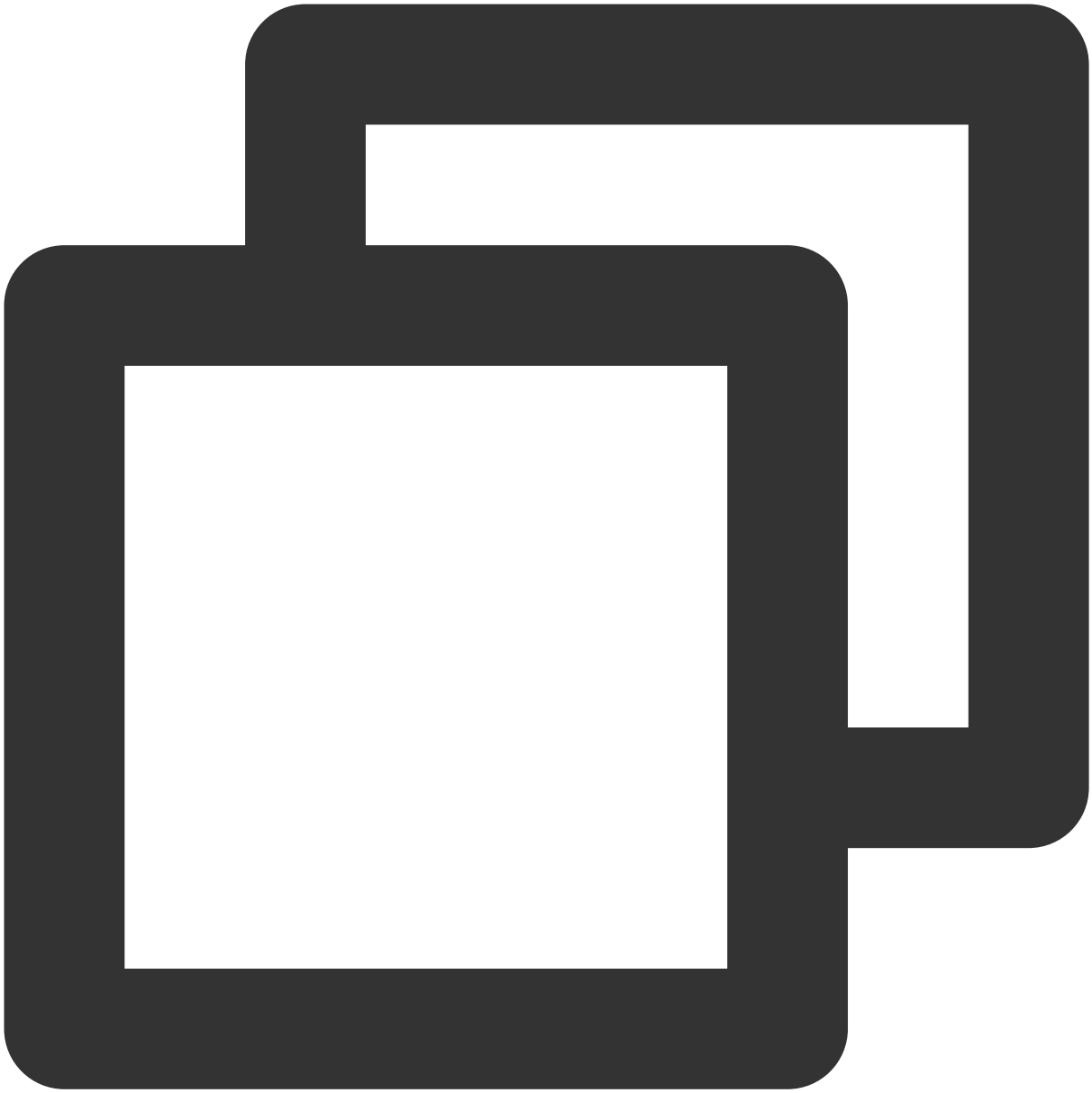
**用户 Tommy 收到的推送效果是**：



```
Hi @Tommy，你的购物车里还有一件企鹅公仔，库存只有6件！快把它买了吧？
```

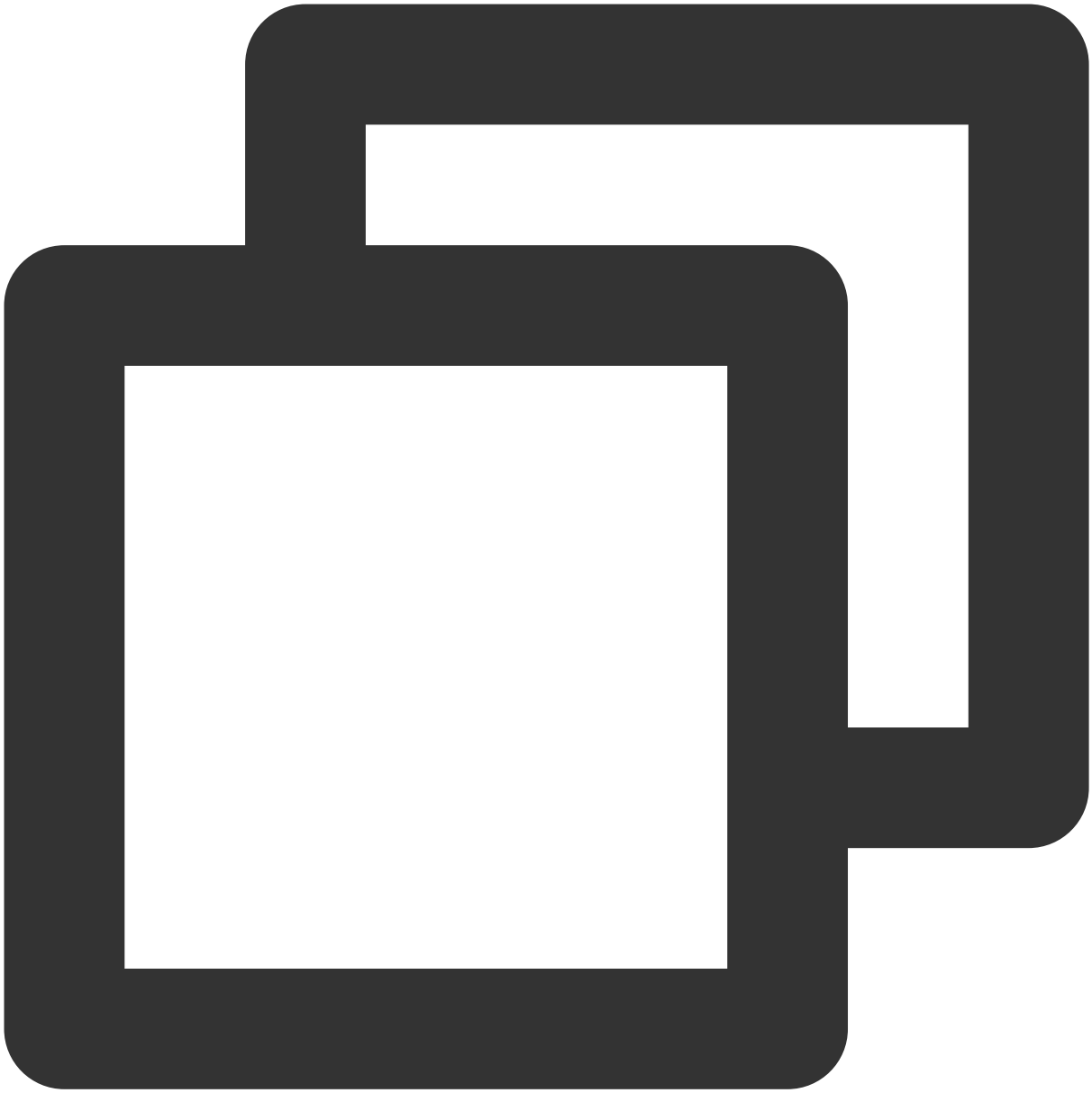
### 游戏应用场景

通过 push 促活沉默游戏玩家，可使用以下模板：



```
Hi @{{nickname}}，你已经{{offline_days}}天没有登录游戏了，我们为你准备了{{gift_num}}份大礼包
快来领取吧>>>
```

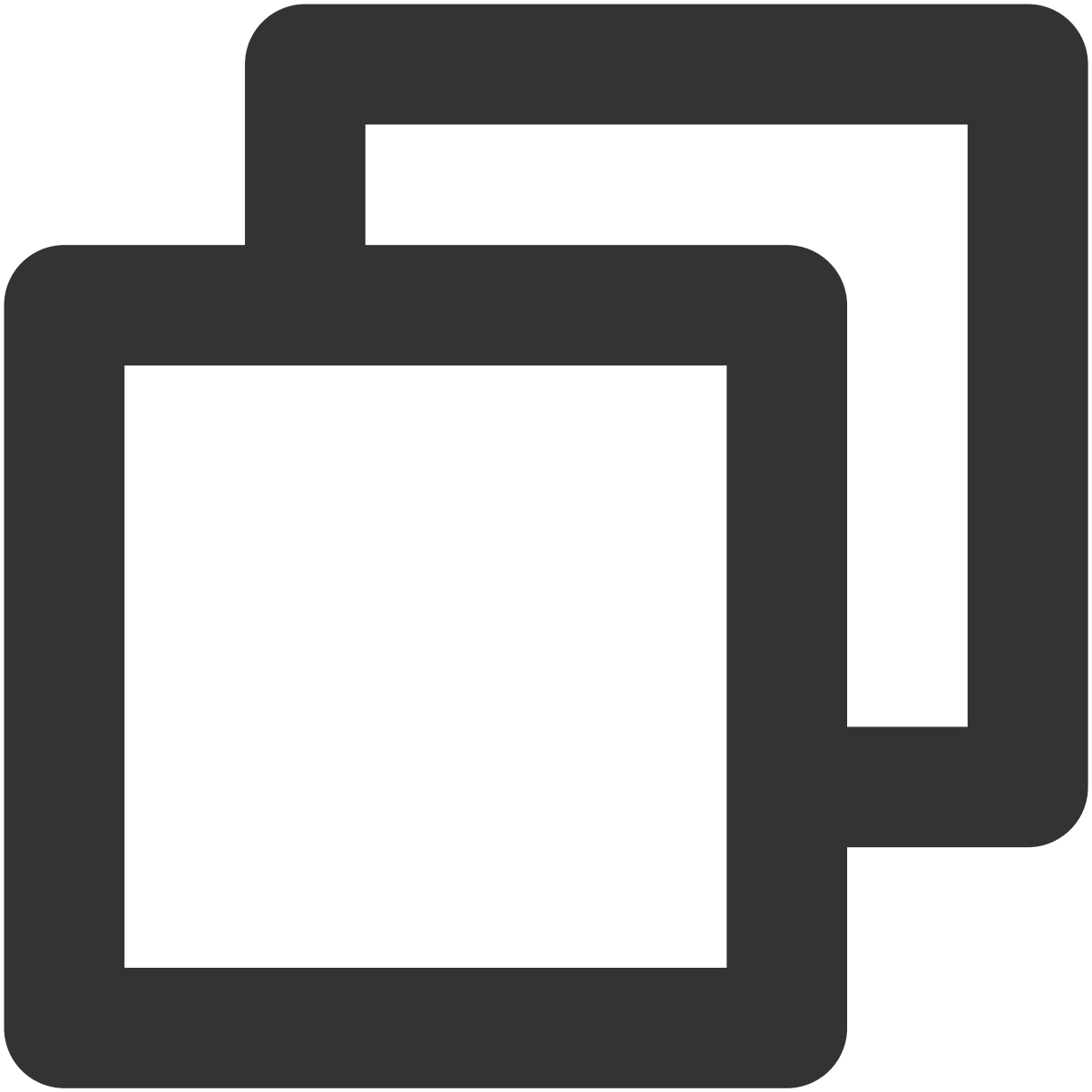
**连续3天未上线的用户 Tommy 收到的推送效果是：**



```
Hi @Tommy，你已经3天没有登录游戏了，我们为你准备了6份大礼包，快来领取吧>>>
```

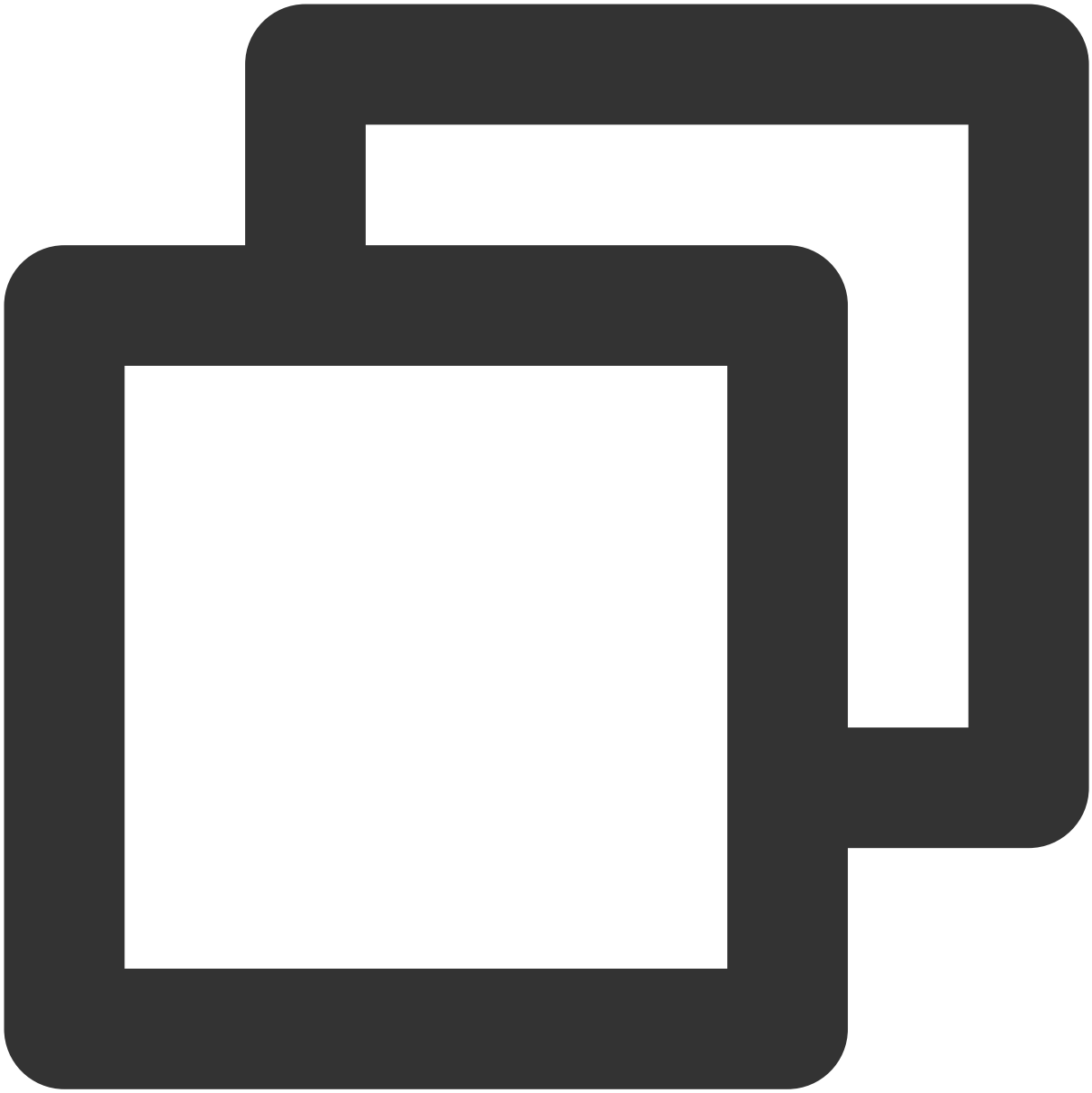
### 社群类应用场景

通过 push 对连续3天未打开应用的用户促活，可使用以下模板：



```
Hi @{{nickname}}，你不在的这些天，你的{{friend_num}}位好友共发布了{{story_num}}条动态，
快来看看吧>>>
```

**用户 Tommy 收到的推送效果是**：



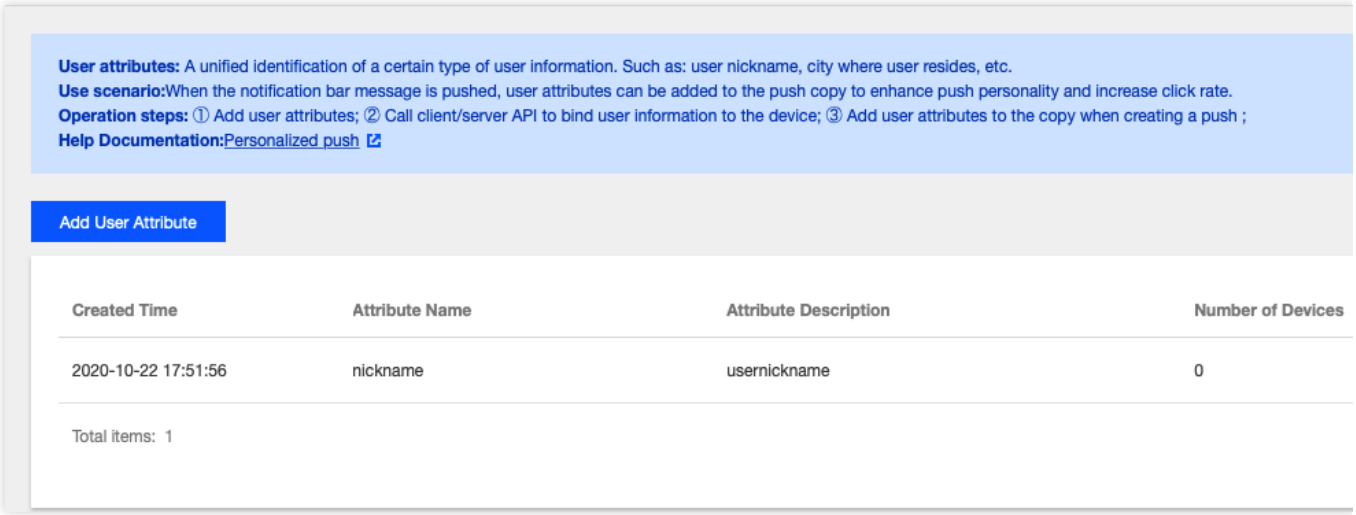
```
Hi @Tommy，你不在的这些天，你的8位好友共发布了20条动态，快来看看吧>>>
```

# 使用条件

### 创建和管理用户属性

1. 登录 移动推送控制台。
2. 前往 **App 推送管理** > **用户属性管理**，单击**新建用户属性**。

3. 在**新建用户属性**对话框中输入属性名称和属性描述后，单击**确定**。创建完成后，可在用户属性管理页查看所有的属性创建时间、属性名称、属性描述、绑定的设备数；并且可随时编辑和删除属性。




### 绑定用户属性

在使用个性化推送前，需要您先将用户属性与设备进行绑定，您可以通过以下2种方式进行绑定：

**方式1：通过客户端接口绑定：**

iOS SDK 进行用户属性的设置，详情请参见 用户属性设置 文档。
Android SDK 进行用户属性的设置，详情请参见 用户属性设置 文档。

**方式2： 通过 Rest API 绑定：**

API 绑定方式可参考 个性化推送相关 API 说明。

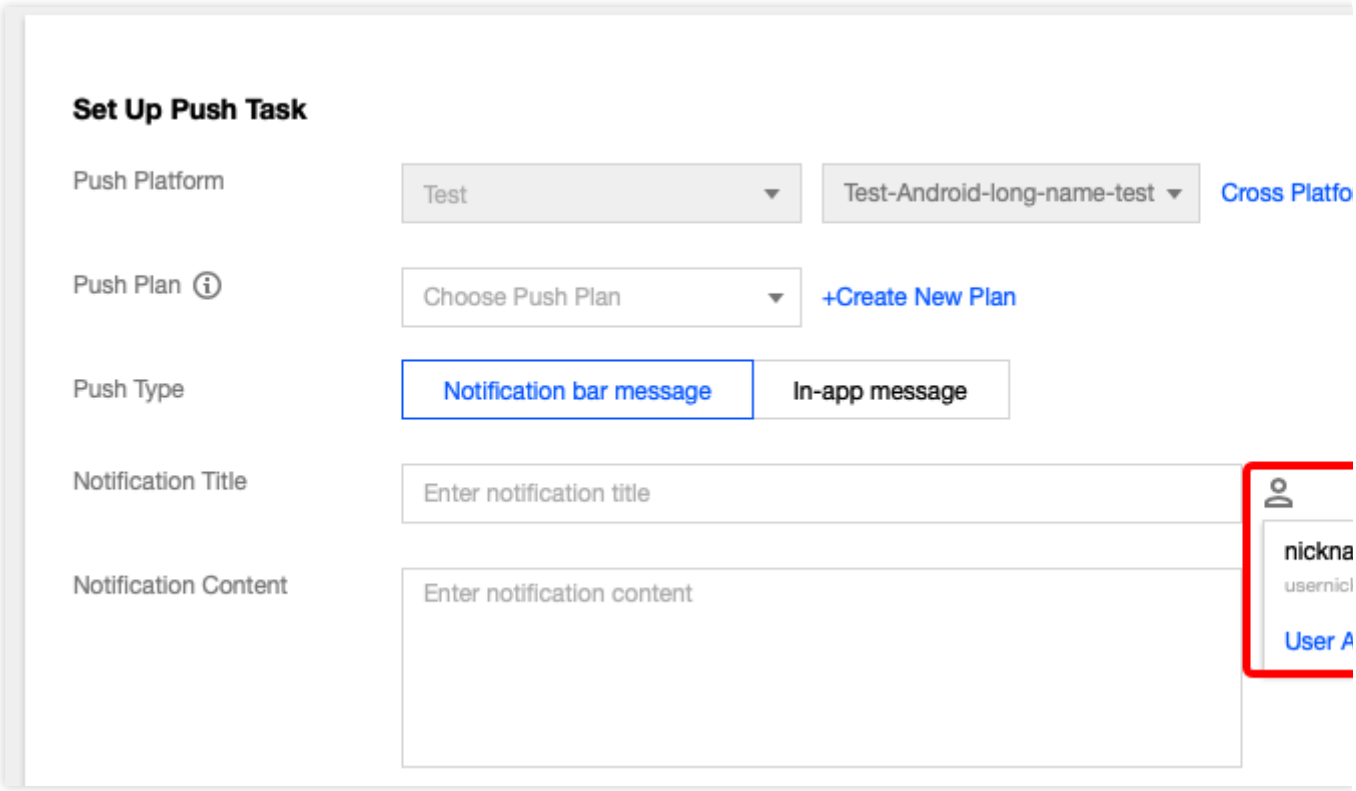
## 开始使用

### 控制台使用

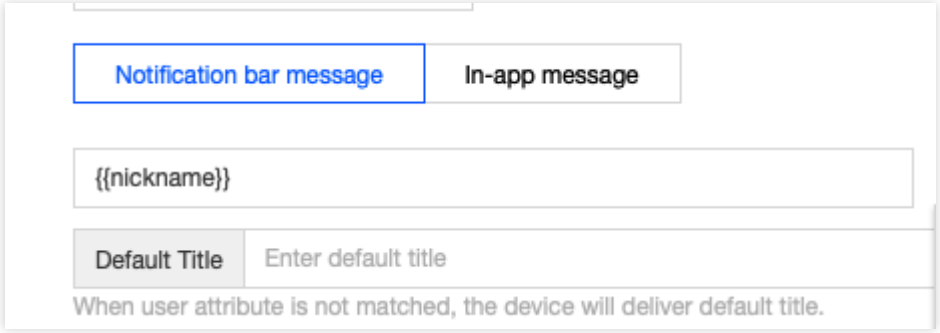
1. 前往 **App 推送管理** > **推送任务**页面，单击新建推送。
2. 在通知标题或通知内容文本框右侧选择插入用户属性。

**说明：**
单条推送文案每次最多可插入5个属性。

版权所有：腾讯云计算（北京）有限责任公司




3. 当用户属性无法匹配时，可设置向用户发送默认展示的标题或内容。




4. 插入完成后，单击**测试预览**，确认信息无误后单击**发送确认**即可完成带用户属性的个性化推送。

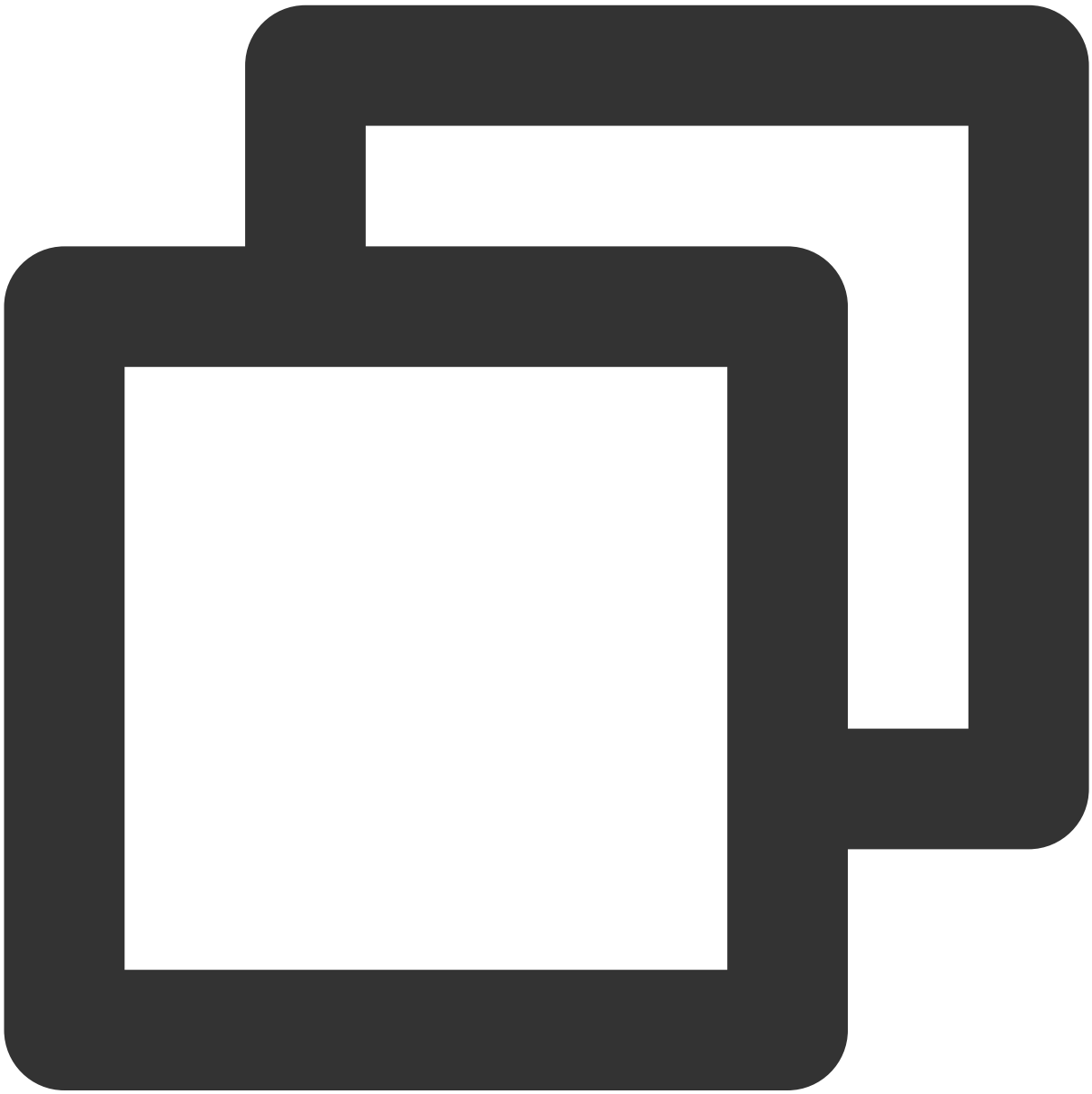
## Rest API 使用

在调用推送接口时，可以通过设置 `ntf_wt_attrs` 为 `ture` 来开启个性化通知，并在 `message` 中补充以下字段：

| 参数名 | 类型 | 是否必须 | 描述 |
| --- | --- | --- | --- |
| default_content | string | 是 | 当用户属性无法匹配时，向设备发送默认消息内容 |
| default_title | string | Android 平台必须，iOS 平台非必须 | 当用户属性无法匹配时，向设备发送默认消息标题 |

| | | | |
|---|---|---|---|
| default_subtitle | string | 否 | 当用户属性无法匹配时，向设备发送默认消息副标题 |

其它消息字段说明可参考 API 消息体说明。
推送示例如下：



```
{
    "audience_type": "token",
    "expire_time": 3600,
    "message_type": "notify",
```

```
    "environment":"dev",
    "message": {
        "title": "你好, {{name}}",
        "content":"你的积分以达到{{score}}",
        "default_content":"默认内容推送",
        "default_title":"默认标题",
        "default_subtitle":"默认副标题"
    },
    "token_list": [
        "086f959c7aefc3****add2ccf0cd539c1edd"
    ],
    "ntf_wt_attrs":true
}
```

版权所有：腾讯云计算（北京）有限责任公司

# 账号推送

最近更新时间：2024-01-16 17:34:39

## 操作场景

运营人员从第三方数据平台或者业务后台获取到一批用户的账号，要对这批用户做运营活动时，可使用 `账号推送` ，一次性向单个或多个用户账号推送。

**注意：**

上述**账号**必须与 移动推送 Token 绑定，操作步骤可参考 账号绑定（Android）或 账号绑定（iOS）接口。

## 操作步骤

### 控制台使用

1. 登录 移动推送控制台。
2. 找到您需要配置的批量推送的应用，在其操作一栏下，选择**新建推送**，进入新建推送页面。
3. 在**推送目标**配置项中，选择**账号**，您可以选择上传号码包文件或手动输入账号，如下图所示：

**说明：**

上传号码包要求如下：

账号包文件名：长度限制为 [1, 100]。

账号包格式及大小： 支持 `zip\\txt\\csv` 文件；大小保持在 100Mb 以内。

`zip` 压缩包中可包含：单个 `.txt` 或 `.csv` 文件；（不能嵌套文件夹）。

. `txt` 文件要求：（1）编码为 UTF-8；（2）每行一个账号，账号长度限制为 [2, 100]。

. `csv` 文件要求：（1）只能有一列；（2）每行一个账号，账号长度限制为 [2, 100]。

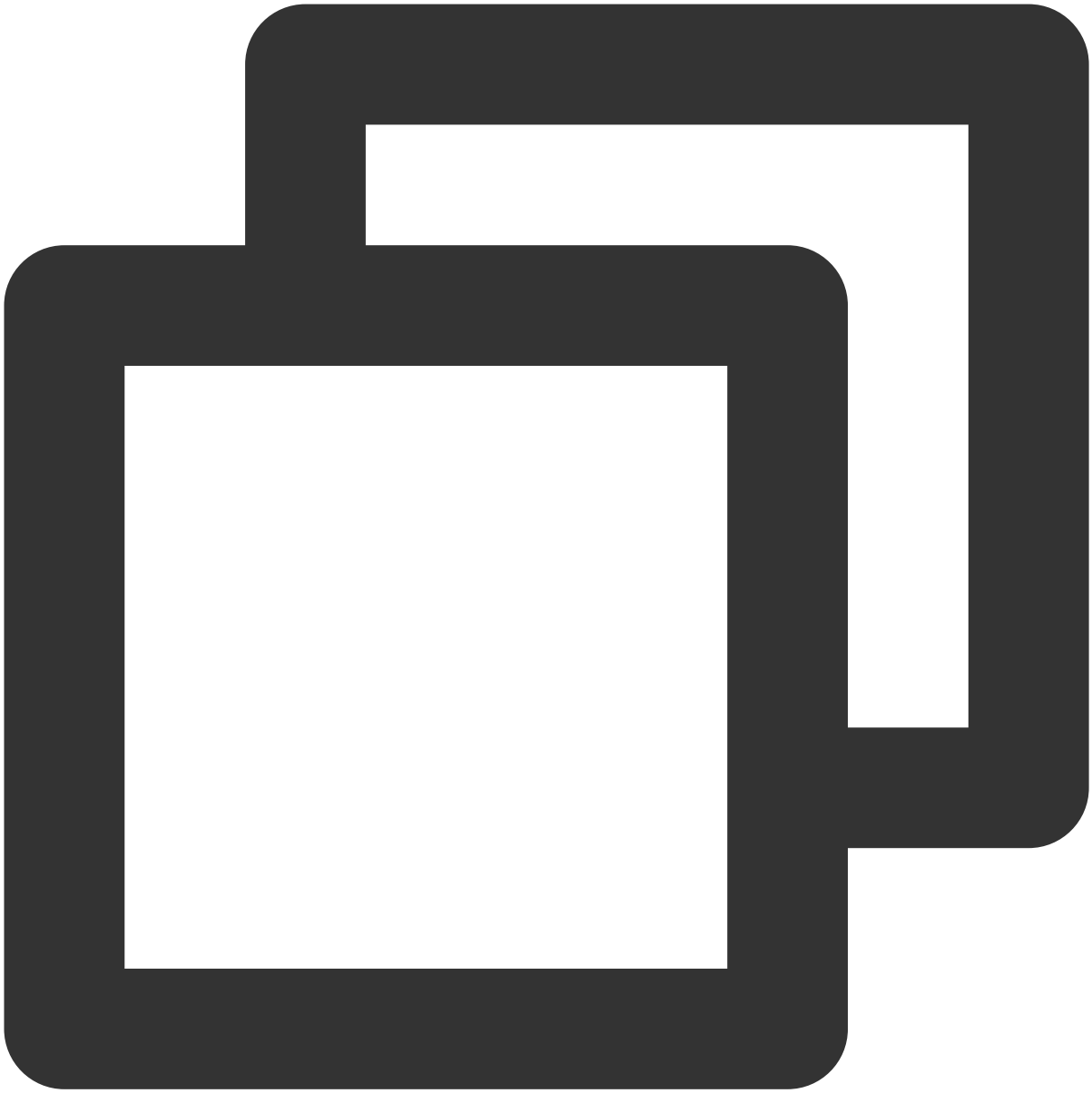
4. 选择账号所对应的**账号类型**，账号类型可咨询业务开发人员获取，如果没有指定，则默认为**默认类型**。
5. 单击**预览**，确认推送配置无误后，单击**确认推送**即可完成按账号推送。

### Rest API 使用

#### 单个/多个账号推送

在您调用 推送接口 时，将 `audience_type` （推送目标）设置为 `account` （单账号推送）或 `account_list` （账号列表推送），并参考 账号类型取值表，填写账号类型所对应的值。

#### 推送示例



```
{
    "audience_type": "account",
        "account_list": [
            "123456"
     ],
    "account_type":1,
    "account_push_type":0,
    "message_type": "notify",
    "message": {
        "title": "恭喜您在本次活动中获奖",
        "content":"快快上线领取您的奖品吧！"
```

```
        }
    }
```

**上传号码包文件推送**

**步骤1：调用上传号码包文件接口**

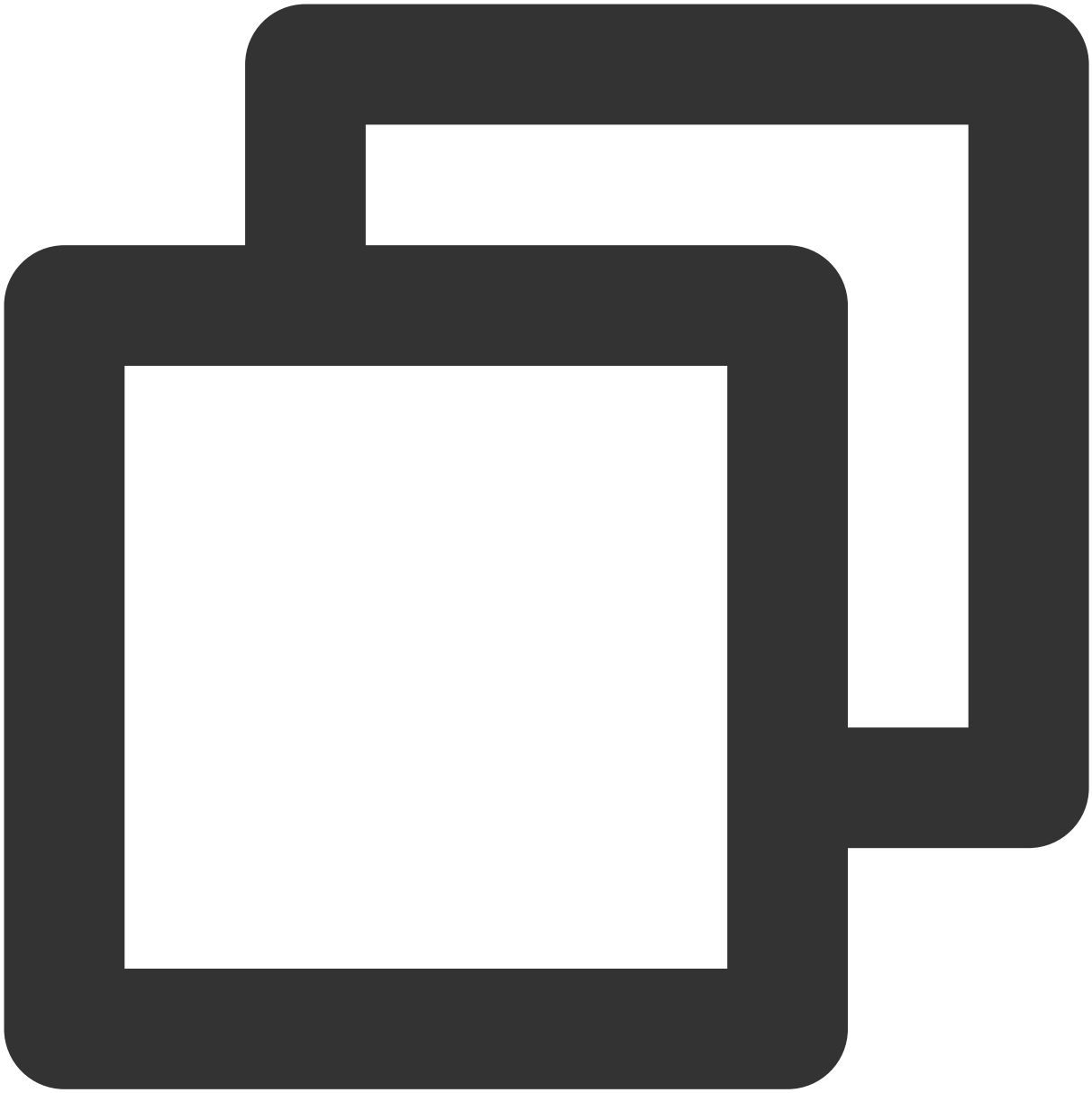
参考 号码包上传接口 文档，上传您的号码包文件，调用成功后会返回 `upload_id` ，例如：11231。

**步骤2：调用推送接口**

1. 调用 推送接口 时，将 `audience_type` （推送目标）设置为 `package_account_push` （号码包推送）。
2. 补充 步骤1 中获取的 `upload_id` ，例如：11231。
3. 参考 账号类型取值表，填写账号类型所对应的值。
4. 设置 `account_push_type` 字段来决定推送给「**最近**」或「**所有**」绑定该账号的设备。

**推送示例**

如下示例为推送给在产品活动中获奖的用户：



```
{
    "audience_type": "package_account_push",
    "upload_id": 11231,
    "account_type":1,
    "account_push_type":0,
    "message_type": "notify",
    "message": {
        "title": "恭喜您在本次活动中获奖",
        "content":"快快上线领取您的奖品吧！"
    }
}
```

版权所有：腾讯云计算（北京）有限责任公司