

云通信

服务端集成

产品文档



腾讯云

【版权声明】

©2013-2018 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模式的承诺或保证。

文档目录

服务端集成

服务端集成指引

REST API 概览

REST API接口列表

账号管理

独立模式账号导入接口

独立模式帐号批量导入接口

帐号登录态失效接口

托管模式存量账号导入

单聊消息

单发单聊消息

批量发单聊消息

导入单聊消息

消息推送

推送

获取推送报告

设置应用属性名称

获取应用属性名称

设置用户属性

删除用户属性

获取用户属性

添加用户标签

获取用户标签

删除用户标签

删除用户所有标签

计费规则

在线状态

获取用户在线状态

资料管理

拉取资料

设置资料

关系链管理

添加好友

导入好友

更新好友

- 删除好友
- 删除所有好友
- 校验好友
- 拉取好友
- 拉取指定好友
- 添加黑名单
- 删除黑名单
- 拉取黑名单
- 校验黑名单
- 添加分组
- 删除分组

群组管理

- 获取APP中的所有群组
- 创建群组
- 获取群组详细资料
- 获取群组成员详细资料
- 修改群组基础资料
- 增加群组成员
- 删除群组成员
- 修改群成员资料
- 解散群组
- 获取用户所加入的群组
- 查询用户在群组中的身份
- 批量禁言和取消禁言
- 获取群组被禁言用户列表
- 在群组中发送普通消息
- 在群组中发送系统通知
- 群组消息撤回
- 转让群组
- 导入群基础资料
- 导入群消息
- 导入群成员
- 设置成员未读消息计数
- 删除指定用户发送的消息
- 拉取群漫游消息

脏字管理

- 查询APP自定义脏字
- 添加APP自定义脏字

删除APP自定义脏字

全局禁言管理

设置全局禁言

查询全局禁言

数据下载

消息记录下载

Server SDK集成

PHP Server SDK

Node.js Server SDK

运营相关

监控配置

拉取运营数据

服务端集成

服务端集成指引

最近更新时间：2018-06-06 14:30:23

内容提要

[服务端集成](#) 是 App 后台通过云通讯提供的一系列 REST API 来管理自身 App 的数据。云通信提供的 REST API 参见 [REST API接口列表](#)。

本文档将介绍如何通过 REST API 来进行数据管理，您将了解到：

1. 调用 REST API 需要哪些前提条件；
2. 如何调试 REST API，包括：
 - 2.1 通过云通讯 REST API 调试工具来调用 REST API；
 - 2.2 通过 Postman 来调试 REST API；
 - 2.3 通过 PHP Server SDK 中的工具来调试 REST API；
3. 如何在服务端调用 REST API，包括：
 - 3.1 在 PHP 中集成 PHP Server SDK。

REST API

REST API 是腾讯云暴露给 App 后台的 HTTP 管理接口，其主要目的在于为 App 后台提供一个后台管理入口。更多介绍参见 [REST API简介](#)。

前提条件

要调用 REST API，您必须首先

1. 在云通信中创建 App，具体方法参见 [应用接入指引](#)；
2. 为您的 App 指定管理员帐号，具体方法参见 [设置App管理员帐号](#)。

特别提醒：调用 REST API 时请务必使用 App 管理员帐号，否则会导致不必要的调用错误。

调用方法

REST API 的请求 URL、请求方式、包体格式参见 [REST API简介](#)。目前支持的 REST API 参见 [REST API接口列表](#)。

本文档将介绍如下几种方式来调用 REST API，分别是：

1. 使用云通讯 REST API 调试工具；
2. 使用 Postman；
3. 使用 PHP Server SDK 中的 REST API 调试工具。

在“调用示例”一节中，将通过具体的案例来演示调用 REST API 的各种方法。

调用示例

本文档将通过一个完整的示例来展示 REST API 的各种调用方法。该示例包含以下步骤：

1. 导入帐号到云通信（使用云通讯 REST API 调试工具）；
2. 为导入的设置帐号的基本资料（使用 Postman）；
3. 创建一个群组，包含初始群成员（使用 PHP Server SDK 中的工具）。

准备工作：获取 App 管理员帐号的签名

确保 App 管理员帐号存在。打开 App 控制台，单击【应用配置】：

创建时间 ▾	操作
2015-06-04 19:16:50	应用配置 接入服务 更多 ▾

进入页面之后，查看【帐号体系集成】，确保后续过程均使用 App 管理员帐号（此处假定 App 管理员帐号的为 myadmin）：

帐号体系集成 [编辑](#)

集成自有帐号体系

帐号名称	ZXC
accountType	183
集成模式	独立模式
验证方式	系统生成公私钥 什么是公私钥

[下载公私钥](#)

系统生成的公私钥可供用户测试服务，每次下载将会生成唯一有效公私钥，为了更好的保障业务安全建议正式上线时使用自有生成公私钥用于验证支持。

帐号管理员 myadmin [什么是帐号管理员](#)

为 App 管理员帐号生成 usersig :

1. Linux 下生成 usersig 的方式参见：[linux 下生成 sig 和校验 sig](#)。
2. Windows 下生成 usersig 的方式参见：[windows 下生成 sig 和校验 sig](#)。

例如，在 Windows 下，生成命令是：

```
.\tls_licence_tools.exe gen .\keys\private_key sig 3600 1400000478 myadmin
```

生成 usersig 之后，便完成了调用 REST API 所需的准备工作。

导入帐号到云通信（使用云通讯 REST API 调试工具）

下面将使用 [云通讯 REST API 调试工具](#) 来导入一个帐号，用户名为 lilei。

（需要导入该帐号的原因是：在后续内容中我们为 lilei 设置资料，并以其为群主来创建一个群，但目前该帐号尚不存在。因此我们需要把 lilei 这个帐号导入到云通讯中。）

打开 [云通讯 REST API 调试工具](#)，填写调用参数如下：

此工具旨在帮助开发者进行云通讯REST API联调，填写相关参数，提交请求后，即可看到请求包和应答包。

SdkAppid:	<input type="text" value="1400000478"/>	请填写在云通讯服务中申请的AppID。
Identifier:	<input type="text" value="myadmin"/>	请填写APP管理员帐号。
UserSig:	<input type="text" value="eJx1kEtPg0AURvf8CsLayJShPExcUEq1wdq0BUu6lWOZaa*"/>	请填写APP管理员帐号的签名。
API:	<input type="text" value="v4/im_open_login_svc/account_import"/> <input type="checkbox"/> 其它	请选择需要测试的API接口。

请求包体:请求包体为JSON格式:

```
{
  "Identifier": "lilei"
}
```

按照你填写的信息，请求包和应答包体如下：

单击【提交】，可以看到应答包体如下：

应答包体:

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

至此，便完成 lilei 这个帐号的导入。

为导入的设置帐号的基本资料（使用 Postman）

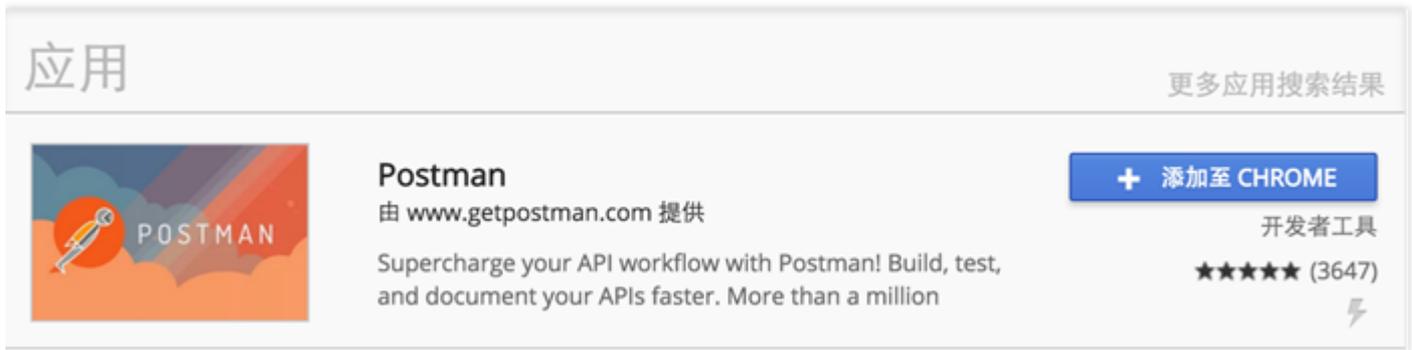
本节我们将使用 [Postman](#) 来调用设置用户资料的接口（v4/profile/portrait_set）为上一节导入的帐号设置基本资料，具体工作是：将用户 lilei 的昵称设置为“李雷”。

安装 Postman

注：安装 Postman 需要能够访问 google 应用商店，如果您的网络无法访问 google 相关网站，建议您使用云通讯 REST API 调试工具。

Postman 是一款功能强大的网页调试与发送网页 HTTP 请求的 Chrome 插件。其安装方法是：打开 Chrome 浏览器，在 Chrome 网上应用商店中搜索 Postman，并【添加至Chrome】，如下图所示：

安装完毕后，Postman 将出现在 Chrome 的应用中，如下图所示：

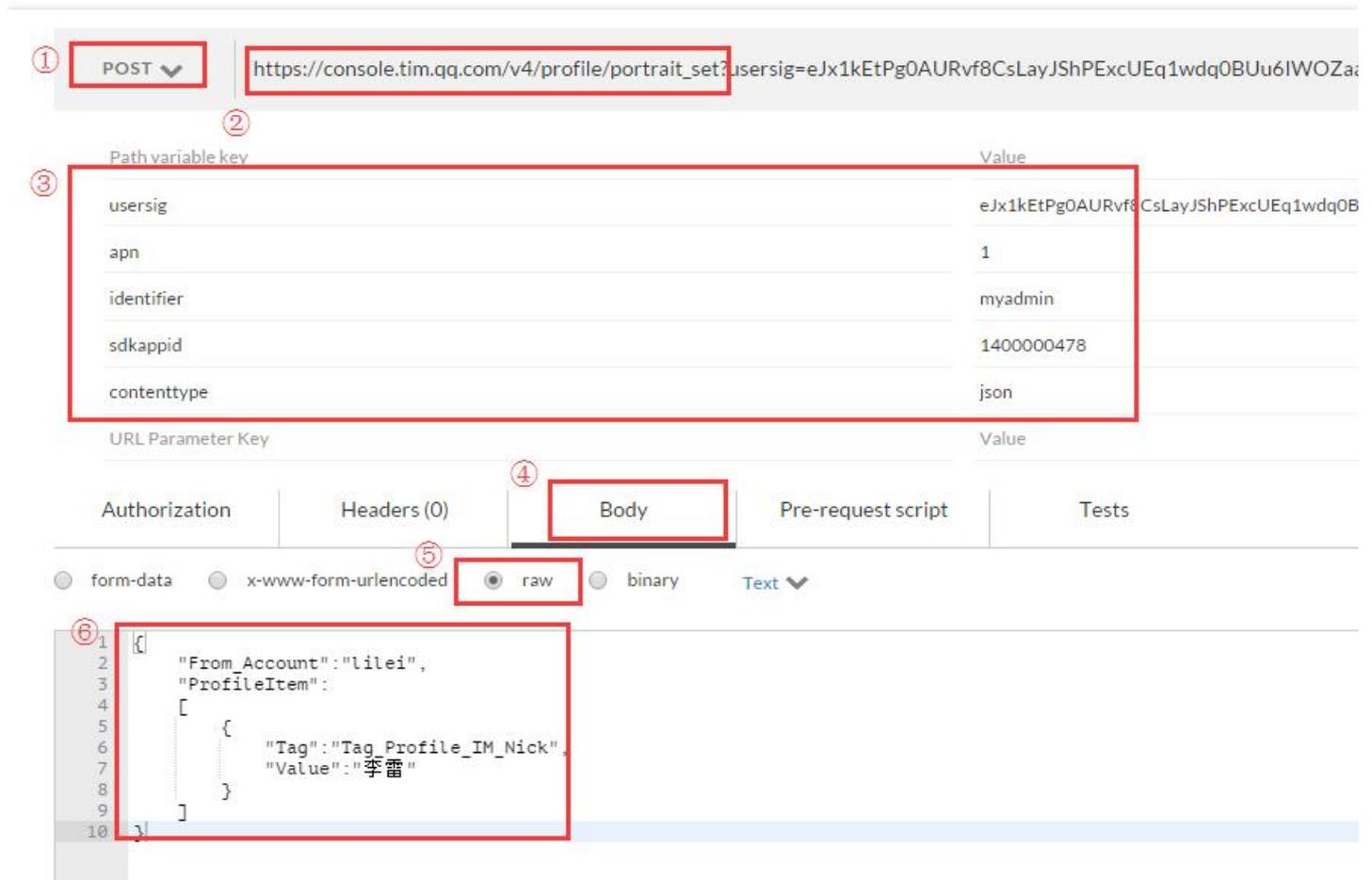


设置用户的基本资料

打开 Postman，依照如下指引填写参数：

1. 选择请求方式为 POST；
2. 在地址栏中填写请求 URL：`https://console.tim.qq.com/v4/profile/portrait_set`；
3. 单击地址栏右侧的“Params”按钮，在下方的参数列表中填写各项请求参数，各参数的含义参见 REST API 参数说明。
4. 单击 Body；
5. 单击 raw，目的是填写文本请求包（JSON 格式）；
6. 填写请求包。

如下图所示：



完成所有请求参数的填写之后，单击右上角的“Send”按钮发起请求，将会收到如下应答（注意将红框处选择JSON）：



至此，我们便实现了通过 Postman 调用 REST API 来设置用户资料。

创建一个群组，包含初始群成员（使用 PHP Server SDK 中的工具）

本节我们将使用 [PHP Server SDK](#) 来调用创建群组的接口（v4/group_open_http_svc/create_group），具体工作是：创建一名字为“MyFirstGrouplilei”的群，群主设置为“lilei”。

[PHP Server SDK](#) 是云通讯 PHP 后台服务提供的 REST API 集成 SDK，该 SDK 将一些常用的 REST API 封装成了 PHP 函数。PHP Server SDK 的下载与集成方法参见：[PHP Server SDK](#)。

完成 PHP Server SDK 的下载与部署后，需要修改 TimRestApiConfig.json 中的 REST API 请求参数：

```
# cat TimRestApiConfig.json
{
  "sdkappid":"1400000478",
  "account_type":"183",
  "identifier":"myadmin",
  "private_pem_path":"./private_key"
```

其中 private_pem_path 为 App 私钥文件的路径。

调用如下命令创建群：

```
php TimRestApiGear.php group_open_http_svc create_group Public MyFirstGroup lilei
```

输出如下：

```
Response Body:
{
  "ActionStatus":"OK",
  "ErrorCode":"0",
  "GroupId":"@TGS#2YJNRPAEZ"
}
```

至此，我们便通过创建了一个群，我们可以在 App 控制台的“群组管理”页中看到该群：

基础配置 SPEAR引擎配置

[+ 添加群组](#) [发送消息](#) [删除群组](#)

<input type="checkbox"/>	群ID	群名称	群主	类型
<input type="checkbox"/>	@TGS#2YJNRPAEZ	MyFirstGroup	lilei	公开群

在后台服务中调用 REST API

在后台服务中调用 REST API，本质上是发起 HTTPS POST 请求。云通信提供了 Server SDK 来封装对 REST API 的调用，开发者可以将其直接集成到您的服务端代码中。

1. [PHP Server SDK](#) ;
2. [Node.js Server SDK](#) ;
3. Java Server SDK (敬请期待) ;
4. Golang Server SDK (敬请期待) 。

REST API 概览

最近更新时间：2018-07-25 11:16:30

REST API 简介

REST API 是腾讯云提供给 App 后台的 HTTP 管理接口，其主要目的在于为 App 后台提供一个后台管理入口。目前云通信支持的 REST API 参见 [REST API 接口列表](#)。

除了 REST API，App [控制台](#) 也可实现简单的数据管理、单发/群发消息，开发者可以在控制台上进行简单的数据管理、查看及测试。相比之下，REST API 接口较为原始，但管理能力却更为强大。

为了安全性，REST API 仅提供 HTTPS 接口。

调用示例

以下为通过 REST API 来获取 App 中所有群组的示例。

HTTPS 请求：

```
POST /v4/group_open_http_svc/get_appid_group_list?accesstoken=xxx&identifier=group_root&sdka  
ppid=88888888&random=99999999&contenttype=json HTTP/1.1
```

Host: console.tim.qq.com

Content-Length: 22

```
{  
  "Limit": 2  
}
```

HTTPS应答：

```
HTTP/1.1 200 OK
```

```
Server: nginx/1.7.10
```

```
Date: Fri, 09 Oct 2015 02:59:55 GMT
```

```
Content-Length: 156
```

```
Connection: keep-alive
```

```
Access-Control-Allow-Origin: *
```

```
Access-Control-Allow-Headers: X-Requested-With
```

```
Access-Control-Allow-Methods: POST
```

```
{  
  "ActionStatus": "OK",  
  "ErrorCode": 0,  
}
```

```

"GroupIdList": [
{
"GroupId": "@TGS#1YTTZEAEG"
},
{
"GroupId": "@TGS#1KVTZAEZ"
}
],
"TotalCount": 58530
}
    
```

调用方法

请求 URL

REST API 的 URL 格式如下：

```

https://console.tim.qq.com/$ver/$servicename/$command?sdkappid=$sdkappid&identifier=$identifier&usersig=$usersig&random=99999999&contenttype=json
    
```

其中各个参数的含义以及取值如下（参数名称及其取值均区分大小写）：

参数	含义	取值
ver	协议版本号。	固定为 v4。
servicename	内部服务名，不同的 servicename 对应不同的服务类型。	参见 API 详细描述。
command	命令字，与 servicename 组合用来标识具体的业务功能。	参见 API 详细描述。
sdkappid	App 在云通信控制台上获取的 Appid。	在申请接入时获得。
identifier	用户名，调用 REST API 时一般为 App 管理员帐号。	用户名（必须为 App 管理员帐号 ）。
usersig	用户名对应的签名。	对于使用独立帐号体系的 App，参见 Linux 平台下生成 usersig 和 Windows 平台下生成 usersig 。对于使用托管帐号体系的 App，参见 下载 UserSig 。

参数	含义	取值
random	标识当前请求的整数随机数参数。	32 位无符号整数随机数。

注意：

1. App 服务端在调用 REST API 时，identifier 必须为 [App 管理员帐号](#)；
2. App 可以在每次调用 REST API 时都生成管理员帐号的 usersig，亦可生成一个固定的 usersig 重复使用，但请特别注意 usersig 的有效期。

HTTP 请求包体格式

REST API 仅支持 POST 方法，其请求包体为 JSON 格式，具体的包体格式参见每个 API 的详细描述。

需要特别注意的是，POST 包体不能为空，即使某条协议包体中不需要携带任何信息，也需要携带一个空的 JSON 对象，即 {}。

HTTP 返回码

除非发生网络错误（例如 502 错误），REST API 的调用结果均为 200，真正的 API 调用错误码与错误信息是在 HTTP 应答包体中返回的。

HTTP 应答包体格式

REST API 的应答包体也是 JSON 格式，其格式符合如下特征：

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  // REST API 其他应答内容
}
```

应答包体中必然包含 ActionStatus、ErrorInfo、ErrorCode 这三个属性，其含义如下：

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败，如果为 FAIL，ErrorInfo 带上失败原因。
ErrorInfo	String	失败原因。
ErrorCode	Integer	错误码，0 为成功，其他为失败，可查询 错误码表 得到具体的原因。

REST API 公共错误码

错误码	含义说明
80001	消息文本安全打击。
60002	HTTP 解析错误，请检查 HTTP 请求 URL 格式。
60003	HTTP 请求 JSON 解析错误，请检查 JSON 格式。
60004	请求 URL 或 JSON 包体中帐号或签名错误。
60005	请求 URL 或 JSON 包体中帐号或签名错误。
60006	appid 失效，请核对 appid 有效性。
60007	rest 接口调用频率超过限制，请降低请求频率。
60008	服务请求超时或 HTTP 请求格式错误，请检查并重试。
60009	请求资源错误，请检查请求 URL。
60010	请求需要 App 管理员 权限，请检查接口调用权限。
60011	appid 请求频率超限，请降低请求频率。
60012	REST 接口需要带 sdkappid，请检查请求 URL 中的 sdkappid。
60013	HTTP 响应包 JSON 解析错误。
60014	置换 ID 超时。
60015	请求包体帐号类型错误，请确认帐号为字符串格式。

REST API 调试工具

您可以通过如下几种工具来 REST API 的调试。

REST API 调试工具

通过 [REST API 在线调试工具](#) 调试本接口
使用案例参见 [这里](#)。

Postman

Postman 是一款功能强大的网页调试与发送网页 HTTP 请求的 Chrome 插件。其使用案例参见 [这里](#)。

PHP Server SDK 中的调试工具

PHP Server SDK中包含了一个 REST API 调试工具：TimRestApiGear.php。您可以通过该工具在服务器上发起简单 REST API 调用。使用案例参见 [这里](#)。

Server SDK 集成

PHP Server SDK

PHP Server SDK将一些常用的 REST API 封装成了函数，并以接口类的方式暴露给开发者。

FAQ

REST 请求有概率超时，收不到任何响应

- (1) IM 后台 REST 接口设置的超时时间是 3s，调用方设置的超时时间应该长于 3s。
- (2) telnet yun.tim.qq.com 443 确认能否连接服务端口。
- (3) 使用 curl -G <https://yun.tim.qq.com> 简单测试确认能够收到响应。
- (4) 确认机器的 dns server 配置是内网 dns server，还是公共 dns server。如果是内网 dns server，请确保 dns server 网络出口和本机器网络出口 IP 所在地域运营商一致。
- (5) 建议业务调用方使用“长连接+连接池”模式。

REST API接口列表

最近更新时间：2018-04-17 18:19:24

账号管理

功能说明	接口
独立模式帐号导入	v4/im_open_login_svc/account_import
独立模式帐号批量导入	v4/im_open_login_svc/multiaccount_import
托管模式帐号导入	v4/registration_service/register_account_v1
失效帐号登录态	v4/im_open_login_svc/kick

单聊消息

功能说明	接口
单发单聊消息	v4/openim/sendmsg
批量发单聊消息	v4/openim/batchsendmsg
导入单聊消息	v4/openim/importmsg

消息推送

功能说明	接口
推送	v4/openim/im_push
获取推送报告	v4/openim/im_get_push_report
设置应用属性名称	v4/openim/im_set_attr_name
获取应用属性名称	v4/openim/im_get_attr_name
设置用户属性	v4/openim/im_set_attr

功能说明	接口
删除用户属性	v4/openim/im_remove_attr
获取用户属性	v4/openim/im_get_attr
添加用户标签	v4/openim/im_add_tag
删除用户标签	v4/openim/im_remove_tag
删除用户所有标签	v4/openim/im_remove_all_tags

群组管理

功能说明	接口
获取APP中的所有群组	v4/group_open_http_svc/get_appid_group_list
创建群组	v4/group_open_http_svc/create_group
获取群组详细资料	v4/group_open_http_svc/get_group_info
获取群成员详细资料	v4/group_open_http_svc/get_group_member_info
修改群组基础资料	v4/group_open_http_svc/modify_group_base_info
增加群组成员	v4/group_open_http_svc/add_group_member
删除群组成员	v4/group_open_http_svc/delete_group_member
修改群组成员资料	v4/group_open_http_svc/modify_group_member_info
解散群组	v4/group_open_http_svc/destroy_group
获取用户所加入的群组	v4/group_open_http_svc/get_joined_group_list
查询用户在群组中的身份	v4/group_open_http_svc/get_role_in_group
批量禁言和取消禁言	v4/group_open_http_svc/forbid_send_msg
获取群组被禁言用户列表	v4/group_open_http_svc/get_group_shutted_uin
在群组中发送普通消息	v4/group_open_http_svc/send_group_msg
在群组中发送系统通知	v4/group_open_http_svc/send_group_system_notification

功能说明	接口
转让群组	v4/group_open_http_svc/change_group_owner
导入群基础资料	v4/group_open_http_svc/import_group
导入群消息	v4/group_open_http_svc/import_group_msg
导入群成员	v4/group_open_http_svc/import_group_member
设置成员未读消息计数	v4/group_open_http_svc/set_unread_msg_num
删除指定用户发送的消息	v4/group_open_http_svc/delete_group_msg_by_sender
拉取群漫游消息	v4/group_open_http_svc/group_msg_get_simple

资料管理

功能说明	接口
拉取资料	v4/profile/portrait_get
设置资料	v4/profile/portrait_set

关系链管理

功能说明	接口
添加好友	v4/sns/friend_add
导入好友	v4/sns/friend_import
删除好友	v4/sns/friend_delete
删除所有好友	v4/sns/friend_delete_all
校验好友	v4/sns/friend_check
拉取好友	v4/sns/friend_get_all
拉取指定好友	v4/sns/friend_get_list
添加黑名单	v4/sns/black_list_add

删除黑名单	v4/sns/black_list_delete
拉取黑名单	v4/sns/black_list_get
校验黑名单	v4/sns/black_list_check
添加分组	v4/sns/group_add
删除分组	v4/sns/group_delete

脏字管理

功能说明	接口
查询脏字	v4/openim_dirty_words/get
添加脏字	v4/openim_dirty_words/add
删除脏字	v4/openim_dirty_words/delete

数据下载

功能说明	接口
消息记录下载	v4/open_msg_svc/get_history

在线状态

功能说明	接口
获取用户在线状态	v4/openim/querystate

全局禁言管理

功能说明	接口
设置全局禁言	v4/openconfigsvr/setnospeaking

功能说明	接口
查询全局禁言	v4/openconfigsvr/getnospeaking

账号管理

独立模式账号导入接口

最近更新时间：2017-11-18 14:48:25

1 功能说明

本接口适用于将应用自有帐号信息导入腾讯云。该接口的功能是在腾讯内部创建一个内部ID，使没有登录腾讯云的应用自有帐号能够使用腾讯云服务。

举个例子：APP开发者通过腾讯云服务给APP用户A发送一条消息，用户A如果没有登录过云通信服务，由于腾讯内部没有用户A对应的内部ID，那么给用户A发送消息将会失败。此时需要把用户A的帐号同步给腾讯，在腾讯内部会为用户A创建一个内部ID，这样就可以给用户A发送消息了。

说明：开发者可根据自身业务场景考虑是否需要调用该接口，接口支持一次同步一个帐号，批量帐号导入请[查看](#)。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/im_open_login_svc/account_import?usersig=xxx&identifier=admin&sd  
kappid=88888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{
  "Identifier": "test",
  "Nick": "test",
  "FaceUrl": "http://www.qq.com"
}
```

2.7 请求包字段说明

字段	类型	属性	说明
Identifier	String	必填	用户名，长度不超过 32 字节
Nick	String	选填	用户昵称
FaceUrl	String	选填	用户头像URL。
Type	Integer	选填	帐号类型，开发者默认无需填写，值0表示普通帐号，1表示机器人帐号。

2.8 应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

2.10 错误码说明

除非发生网络错误(例如502错误),该接口的HTTP返回码均为200。真正的错误码、错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码 (60000到79999) 参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
70402	参数非法。请检查必填字段是否填充，或者字段的填充是否满足协议要求。
70403	发起操作者不是 APP管理员 ，没有权限操作。
70404	设置简单资料后端超时。

3 接口调试工具

3.1 Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4 可能触发的回调

无。

5 参考

无。

独立模式帐号批量导入接口

最近更新时间：2017-11-18 14:48:47

1 功能说明

本接口适用于将应用自有帐号信息批量导入腾讯云。该接口的功能是在腾讯内部创建一个内部ID，使没有登录腾讯云的应用自有帐号能够使用腾讯云服务。

举个例子：APP开发者通过腾讯云服务给APP用户A发送一条消息，用户A如果没有登录过云通信服务，由于腾讯内部没有用户A对应的内部ID，那么给用户A发送消息将会失败。此时需要把用户A的帐号同步给腾讯，在腾讯内部会为用户A创建一个内部ID，这样就可以给用户A发送消息了。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/im_open_login_svc/multiaccount_import?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

该接口最大调用频率为10次/秒。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{
  "Accounts":["test1","test2","test3","test4","test5"]
}
```

2.7 请求包字段说明

字段	类型	属性	说明
Accounts	Array	必填	用户名，单个用户名长度不超过 32 字节，单次最多导入100个用户名

2.8 应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "",
  "FailAccounts": [
    "test3",
    "test4"
  ]
}
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
FailAccounts	Array	导入失败的帐号列表

2.10 错误码说明

除非发生网络错误(例如502错误),该接口的HTTP返回码均为200。真正的错误码、错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码 (60000到79999) 参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
70402	参数非法。请检查必填字段是否填充，或者字段的填充是否满足协议要求。
70403	发起操作者不是APP管理员，没有权限操作。

3 接口调试工具

3.1 Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4 可能触发的回调

无。

5 参考

无。

帐号登录态失效接口

最近更新时间：2017-11-08 22:39:09

1 功能说明

本接口适用于将应用帐号的登录态（比如UserSig）失效。

比如：开发者判断一个用户为恶意帐号后，可以调用本接口将该用户当前的登录态失效。

说明：开发者可根据自身业务场景考虑是否需要调用该接口，接口支持一次失效一个帐号。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/im_open_login_svc/kick?usersig=xxx&identifier=admin&sdkappid=8888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

1000次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{
  "Identifier": "test"
}
```

2.7 请求包字段说明

字段	类型	属性	说明
Identifier	String	必填	用户名

2.8 应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

2.10 错误码说明

除非发生网络错误(例如502错误),该接口的HTTP返回码均为200。真正的错误码、错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码 (60000到79999) 参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
70402	参数非法。请检查必填字段是否填充，或者字段的填充是否满足协议要求。
70403	发起操作者不是APP管理员，没有权限操作。

3 接口调试工具

3.1 Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4 可能触发的回调

无。

5 参考

无。

托管模式存量账号导入

最近更新时间：2017-11-08 22:41:05

1 功能说明

此接口用于托管模式下将用户的存量帐号导入到帐号系统。

说明：此接口支持开发者一次导入一个帐号，目前暂无批量帐号导入接口。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/registration_service/register_account_v1?usersig=xxx&identifier=admi  
n&sdkappid=88888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

1000次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{  
  "Identifier":"testaccount1",  
  "IdentifierType":3,  
  "Password":"pas122333"  
}
```

2.7 请求包字段说明

字段	类型	属性	说明
Identifier	String	必填	为用户申请同步的帐号，长度为4-24个字符。
IdentifierType	Integer	必填	Identifier的类型，1:手机号(国家码-手机号) 2:邮箱 3:字符串帐号
Password	String	必填	Identifier的密码，长度为8-16个字符。

注：

字符串帐号忽略大小写，由a-z,0-9及'.'和'_'组成，且不能全为数字。

2.8 应答包体示例

```
{
  "ActionStatus":"OK",
  "ErrorInfo": "",
  "ErrorCode":0
}
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误码信息。

2.10 错误码说明

除非发生网络错误(例如502错误),该接口的HTTP返回码均为200。真正的错误码、错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码 (60000到79999) 参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
70402	已经有此帐号，导入失败。
70405	导入失败，请稍后重试。比如网络超时等，可通过重试解决。
70411	sdkappid非法，比如sdkappid不存在。

错误码	含义说明
70418	请求包格式错误。请检查必填字段是否填充，或者字段的填充是否满足协议要求。
70419	Identifier长度非法。Identifier合法长度为4-24个字符。
70421	Identifier不满足IdentifierType指定的类型。IdentifierType类型为，1:手机号(国家码-手机号) 2:邮箱 3:字符串帐号。
70425	发起操作者不是 APP管理员 ，没有权限操作。

3 接口调试工具

3.1 Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4 可能触发的回调

无。

5 参考

无。

单聊消息

单发单聊消息

最近更新时间：2018-08-24 17:06:36

功能说明

1. 管理员向帐号发消息，接收方看到消息发送者是管理员；
2. 管理员指定某一账户向其他账户发消息，接收方看到发送者不是管理员，而是管理员指定的帐号。

注意，当使用服务端集成 REST API 发送单聊消息时，存在是否将消息同步至发送方（管理员账户或者由管理员指定的某账户）问题，同步方式包括在线终端和漫游，REST API 提供 SyncOtherMachine 参数用于说明是否进行同步，详细使用方式参见下文 [请求包示例](#)。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/openim/sendmsg?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

管理员向其它帐号发消息

注意：若不希望将消息同步至 From_Account，则 SyncOtherMachine 填写 2。若希望将消息同步至 From_Account，则 SyncOtherMachine 填写 1。

```
{
  "SyncOtherMachine": 2, //消息不同步至发送方
  "To_Account": "lumotuwe2",
  "MsgLifeTime": 60, //消息保存60秒
  "MsgRandom": 1287657,
  "MsgTimeStamp": 5454457,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员指定某一帐号向其它帐号发送消息，同时设置离线推送信息，并且不将消息同步至 From_Account

注意：若不希望将消息同步至 From_Account，则 SyncOtherMachine填写 2。

```
{
  "SyncOtherMachine": 2,
  "From_Account": "lumotuwe1",
  "To_Account": "lumotuwe2",
  "MsgLifeTime": 3600, //消息保存一小时
  "MsgRandom": 1287657,
  "MsgTimeStamp": 5454457,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

```
}
],
"OfflinePushInfo": {
  "PushFlag": 0,
  "Desc": "离线推送内容",
  "Ext": "这是透传的内容",
  "AndroidInfo": {
    "Sound": "android.mp3"
  },
  "ApnsInfo": {
    "Sound": "apns.mp3",
    "BadgeMode": 1, // 这个字段缺省或者为 0 表示需要计数，为 1 表示本条消息不需要计数，即右上角图标数字不增加
    "Title": "apns title", // apns title
    "SubTitle": "apns subtitle", // apns subtitle
    "Image": "www.image.com" // image url
  }
}
}
```

管理员指定某一帐号向另一帐号发送消息，同时将消息同步到 From_Account 发送方终端

注意：若希望将消息同步至 From_Account，则 SyncOtherMachine 填写 1。

```
{
  "SyncOtherMachine": 1, //消息同步至发送方
  "From_Account": "lumotuwe1",
  "To_Account": "lumotuwe2",
  "MsgRandom": 1287657,
  "MsgTimeStamp": 5454457,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
SyncOtherMachine	Integer	选填	1：把消息同步到 From_Account 在线终端和漫游上； 2：消息不同步至 From_Account；若不填写默认情况下会将消息存 From_Account 漫游
From_Account	String	选填	消息发送方帐号。（用于指定发送消息方帐号）
To_Account	String	必填	消息接收方帐号。
MsgLifeTime	Integer	选填	消息离线保存时长（秒数），最长为 7 天（604800s）。若消息只发在线用户，不想保存离线，则该字段填 0。若不填，则默认保存 7 天
MsgRandom	Integer	必填	消息随机数,由随机函数产生。（用作消息去重）
MsgTimeStamp	Integer	选填	消息时间戳，UNIX 时间戳。
MsgBody	Object	必填	消息内容，具体格式请参考 消息格式描述 。（注意，一条消息可包括多种消息元素，MsgBody 为 Array 类型）
MsgType	String	必填	TIM消息对象类型，目前支持的消息对象包括： TIMTextElem(文本消息),TIMFaceElem(表情消息)， TIMLocationElem(位置消息)，TIMCustomElem(自定义消息)。
MsgContent	Object	必填	对于每种 MsgType 用不同的 MsgContent 格式，具体可参考 消息格式描述 。
OfflinePushInfo	Object	选填	离线推送信息配置，具体可参考 消息格式描述 。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MsgTime": 1497238162
}
```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
MsgTime	Integer	消息时间戳，UNIX 时间戳。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90002	Json 格式请求包中 MsgBody 不符合消息格式描述，或者 MsgBody 不是 Array 类型，请参考 消息格式描述 。
90003	Json 格式请求包中 To_Account 不符合消息格式描述，请检查 To_Account 类型是否为 String。
90004	Json 格式请求包体中 MsgSeq 不是 Number 类型或者其值范围不在 0~4294967295。
90005	Json 格式请求包体中 MsgRandom 不是 Number 类型或者其值范围不在 0~4294967295。
90006	Json 格式请求包体中 MsgTimeStamp 不是 Number 类型或者其值范围不在 0~4294967295。
90007	Json 格式请求包体中 MsgBody 类型不是 Array 类型，请将其修改为 Array 类型。
90008	Json 格式请求包体中 From_Account 解析失败，请检查 From_Account 是否是 String 类型。
90009	非管理员账户，无权代发消息，请用管理员帐号使用该 Rest API 接口。
90010	Json格式请求包不符合消息格式描述，请参考 消息格式描述 的定义。
90011	批量发消息目标账户超过 500，请减少 To_Account 中目标帐号数量。
90012	To_Account 没有注册或不存在，请确认 To_Account 是否导入腾讯云或者是否拼写错误。
90028	To_Account 不是 Array类型。
90029	To_Account 帐号数量为 0，请添加目标账户。

错误码	含义说明
90030	SyncFromOldSystem 不是 Number 类型或者其值范围不在 0~4294967295
90031	SyncFromOtherMachine 不是 Number 类型或者其值范围不在 0~4294967295
91000	服务内部错误，请重试。
90992	服务内部错误，请重试；如果所有请求都返回该错误码，且 App 配置了 第三方回调 ，请检查 App 服务器是否正常向腾讯云后台服务器返回回调结果。
20002	签名验证失败。
20003	目标账目不存在或者帐号未导入腾讯云。
20006	触发 发单聊消息之前回调 ，App 后台返回禁止下发该消息。

接口调试工具

Web调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 单发文本消息
./TimRestApiGear.php openim sendmsg (account_id) (receiver) (text_content)

# 单发图片消息
./TimRestApiGear.php openim sendmsg_pic (account_id) (receiver) (pic_path)
```

更多调试工具参见 [REST API 调试工具](#)。

API集成

PHP集成

在云通信PHP Server SDK中，可以通过TimRestInterface的如下成员函数快速集成该API：

```
// 单发文本消息
abstract function openim_send_msg($account_id, $receiver, $text_content);
```

```
// 单发图片消息
```

```
abstract function openim_send_msg_pic($account_id, $receiver, $pic_path);
```

```
// 单发消息 (高级接口)
```

```
abstract function openim_send_msg2($account_id, $receiver, $msg_content);
```

可能触发的回调

[发单聊消息之前回调](#)；

[发单聊消息之后回调](#)。

参考

[批量发单聊消息 \(v4/openim/batchsendmsg\)](#)；

批量发单聊消息

最近更新时间：2018-04-20 22:31:33

功能说明

1. 该接口支持一次性针对最多 500 个用户进行单发消息；
2. 与单发消息相比，该接口更适用于营销类消息、系统通知 tips 等时效性较强的消息；
3. 管理员指定某一账户向目标账户批量发消息，接收方看到发送者不是管理员，而是管理员指定的帐号；
4. 该接口不触发回调请求。

注意，当使用服务端集成 REST API 发送批量消息时，存在是否将消息同步至发送方（管理员账户或者由管理员指定的某账户）问题，同步方式包括在线终端和漫游，REST API 提供 SyncOtherMachine 参数用于说明是否进行同步，详细使用方式参见下文 [请求包示例](#)。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/batchsendmsg?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 管理员向目标账户批量发消息

注意，若不希望将消息同步至 From_Account，则 SyncOtherMachine 填写 2。若希望将消息同步至 From_Account，则 SyncOtherMachine 填写 1。

```
{
  "SyncOtherMachine": 2, //消息不同步至发送方
  "To_Account": [ //目标账户列表
    "bonnie",
    "rong"
  ],
  "MsgRandom": 19901224, //消息随机数
  "MsgBody": [ //消息
    {
      "MsgType": "TIMTextElem", //消息类型，TIMTextElem为文本消息
      "MsgContent": {
        "Text": "hi, beauty" //消息文本
      }
    }
  ]
}
```

• 管理员指定某一帐号向目标帐号批量发消息，同时设置离线推送信息

From_Account 为管理员指定的发送方，接收方看到发送者不是管理员，而是 From_Account。下述 Json 请求表达的是 dave 向帐号 bonnie 和帐号 rong 发送一条消息。bonnie 和 rong 收到消息，看到消息发送方是 dave。

注意，若不希望将消息同步至 From_Account，则 SyncOtherMachine 填写 2。若希望将消息同步至 From_Account，则 SyncOtherMachine 填写 1。

```
{
  "SyncOtherMachine": 1, //消息同步至发送方
  "From_Account": "dave",
  "To_Account": [
    "bonnie",
    "rong"
  ],
  "MsgRandom": 19901224,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
```

```

    "Text": "hi, beauty"
  }
  },
  "OfflinePushInfo": {
    "PushFlag": 0,
    "Desc": "离线推送内容",
    "Ext": "这是透传的内容",
    "AndroidInfo": {
      "Sound": "android.mp3"
    },
    "ApnsInfo": {
      "Sound": "apns.mp3",
      "BadgeMode": 1, // 这个字段缺省或者为 0 表示需要计数，为 1 表示本条消息不需要计数，即右上角图标数字不增加
      "Title": "apns title", // apns title
      "SubTitle": "apns subtitle", // apns subtitle
      "Image": "www.image.com", // image url
    }
  }
}
}
}

```

请求包字段说明

字段	类型	属性	说明
SyncOtherMachine	Integer	选填	1：把消息同步到From_Account在线终端和漫游上； 2：消息不同步至From_Account；若不填写默认情况下会将消息存 From_Account 漫游
From_Account	String	选填	管理员指定消息发送方帐号。
To_Account	String	必填	消息接收方用户 identifier。
MsgRandom	Integer	必填	消息随机数,由随机函数产生个随机数即可。
MsgBody	Object	必填	TIM 消息，请参考 消息格式描述 。
MsgType	String	必填	TIM 消息对象类型，目前支持的消息对象包括： TIMTextElem(文本消息), TIMFaceElem(表情消息)， TIMLocationElem(位置消息)，TIMCustomElem(自定义消息)。
MsgContent	Object	必填	MsgContent 为 TIM 消息对象，具体可参考 消息格式描述 。
OfflinePushInfo	Object	选填	离线推送信息配置，具体可参考 消息格式描述 。

应答

应答包体示例

```

{
  "ActionStatus": "FAIL",
  "ErrorInfo": "",
  "ErrorList": [ //发送消息失败列表
    {
      "To_Account": "rong", //目标账户
      "ErrorCode": 20003 //错误码
    }
  ]
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorList	Array	发消息失败列表；在此列表中的目标账户，消息发送失败；消息全部发送成功，ErrorList 为空。因此判断回包中是否包含 ErrorList 来判断，消息是否全都判断成功。
To_Account	String	发送消息失败的帐号。
ErrorCode	Number	发送失败的错误码。

错误码说明

除非发生网络错误(例如 502 错误),该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90002	Json 格式请求包中 MsgBody 不符合消息格式描述，或者 MsgBody 不是 Array 类型，请参考 消息格式描述 。
90003	Json 格式请求包中 To_Account 不符合消息格式描述，请检查 To_Account 类型是否为 String。

错误码	含义说明
90004	Json 格式请求包体中 MsgSeq 不是 Number 类型或者其值范围不在 0~4294967295。
90005	Json 格式请求包体中 MsgRandom 不是 Number 类型或者其值范围不在 0~4294967295。
90006	Json 格式请求包体中 MsgTimeStamp 不是 Number 类型或者其值范围不在 0~4294967295。
90007	Json 格式请求包体中 MsgBody 类型不是 Array 类型，请将其修改为 Array 类型。
90008	Json 格式请求包体中 From_Account 解析失败，请检查 From_Account 是否是 String 类型。
90009	非管理员账户，无权代发消息，请用管理员帐号使用该 Rest API 接口。
90010	Json 格式请求包不符合消息格式描述，请参考 消息格式描述 的定义。
90011	批量发消息目标账户超过 500，请减少 To_Account 中目标帐号数量。
90012	To_Account 没有注册或不存在，请确认 To_Account 是否导入腾讯云或者是否拼写错误。
90028	To_Account 不是 Array 类型。
90029	To_Account 帐号数量为 0，请添加目标账户。
90030	SyncFromOldSystem 不是 Number 类型或者其值范围不在 0~4294967295
90031	SyncFromOtherMachine 不是 Number 类型或者其值范围不在 0~4294967295
91000	服务内部错误，请重试。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 PHP Server SDK 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 批量单发文本消息
./TimRestApiGear.php openim batchsendmsg (account_id1,account_id2,...) (text_content)
```

```
# 批量单发图片消息
./TimRestApiGear.php openim batchsendmsg_pic (account_id1,account_id2,...) (pic_path)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 `TimRestInterface` 的如下成员函数快速集成该 API：

```
// 批量单发文本消息  
abstract function openim_batch_sendmsg($account_list, $text_content);  
  
// 批量单发图片消息  
abstract function openim_batch_sendmsg_pic($account_list, $pic_path);  
  
// 批量单发消息（高级接口）  
abstract function openim_batch_sendmsg2($account_list, $msg_content);;
```

参考

单发单聊消息（[v4/openim/sendmsg](#)）。

群组内发普通消息（[v4/group_open_http_svc/send_group_msg](#)）。

在群组中发送系统通知（[v4/group_open_http_svc/send_group_system_notification](#)）。

[消息格式描述](#)。

导入单聊消息

最近更新时间：2018-04-20 22:21:04

功能说明

1. 该 API 接口 `v4/openim/importmsg` 作用是导入历史单聊消息到腾讯云；
2. 平滑过渡期间，将原有即时通信实时单聊消息导入到腾讯云；
3. 该接口不会触发回调。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/openim/importmsg?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请联系商务经理提需求工单。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- 平滑过渡期间，实时消息导入

```
{
  "SyncFromOldSystem": 1, //平滑过渡期间，实时消息导入，消息计入未读
  "From_Account": "lumotuwe1", //消息发送方帐号
  "To_Account": "lumotuwe2", //接收方帐号
  "MsgRandom": 1287657, //消息随机数
}
```

```

"MsgTimeStamp": 5454457, //消息时间, unix时间戳
"MsgBody": [ //消息内容, 本条消息包括两类消息元素, 一个文本和一个表情
{
"MsgType": "TIMTextElem", //文本消息元素
"MsgContent": {
"Text": "hi, beauty"
}
}
]
}
    
```

• 历史消息导入

```

{
"SyncFromOldSystem": 2, //历史消息导入, 消息不计入未读
"From_Account": "lumotuwe1", //消息发送方帐号
"To_Account": "lumotuwe2", //接收方帐号
"MsgRandom": 1287657, //消息随机数
"MsgTimeStamp": 5454457, //消息时间, unix时间戳
"MsgBody": [ //消息内容, 本条消息包括两种消息元素, 一个文本和一个表情
{
"MsgType": "TIMTextElem", //文本消息元素
"MsgContent": {
"Text": "hi, beauty" //消息内容
}
}
]
}
    
```

请求包字段说明

字段	类型	属性	说明
SyncFromOldSystem	Integer	必填	1, 实时消息导入, 消息加入未读计数; 2, 历史消息导入, 消息不计入未读。
From_Account	String	必填	消息发送方帐号。(用于指定发送消息方)
To_Account	String	必填	消息接收方帐号。
MsgRandom	Integer	必填	消息随机数,由随机函数产生。(用作消息去重)
MsgTimeStamp	Integer	选填	消息时间戳, unix时间戳。
MsgBody	Object	必填	消息内容, 具体格式请参考 消息格式描述 。(注意, 一条消息可包括多种消息元素, MsgBody 为 Array类型)

字段	类型	属性	说明
MsgType	String	必填	TIM 消息对象类型，目前支持的消息对象包括： TIMTextElem(文本消息),TIMFaceElem(表情消息)， TIMLocationElem(位置消息)，TIMCustomElem(自定义消息)。
MsgContent	Object	必填	对于每种 MsgType 用不同的 MsgContent 格式，具体可参考 消息格式描述 。

应答

应答包体示例

```
{
  "ActionStatus" : "OK",
  "ErrorInfo" : "",
  "ErrorCode" : 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90002	Json 格式请求包中 MsgBody 不符合消息格式描述，或者 MsgBody 不是 Array 类型，请参考 消息格式描述 。

错误码	含义说明
90003	Json 格式请求包中 To_Account 不符合消息格式描述，请检查 To_Account 类型是否为 String。
90004	Json 格式请求包体中 MsgSeq 不是 Number 类型或者其值范围不在 0~4294967295。
90005	Json 格式请求包体中 MsgRandom 不是 Number 类型或者其值范围不在 0~4294967295。
90006	Json 格式请求包体中 MsgTimeStamp 不是 Number 类型或者其值范围不在 0~4294967295。
90007	Json 格式请求包体中 MsgBody 类型不是 Array 类型，请将其修改为 Array 类型。
90008	Json 格式请求包体中 From_Account 解析失败，请检查 From_Account 是否是 String 类型。
90009	非管理员账户，无权代发消息，请用管理员帐号使用该 Rest API 接口。
90010	Json 格式请求包不符合消息格式描述，请参考 消息格式描述 的定义。
90011	批量发消息目标账户超过 500，请减少 To_Account 中目标帐号数量。
90012	To_Account 没有注册或不存在，请确认 To_Account 是否导入腾讯云或者是否拼写错误。
90028	To_Account 不是 Array 类型。
90029	To_Account 帐号数量为 0，请添加目标账户。
90030	SyncFromOldSystem 不是 Number 类型或者其值范围不在 0~4294967295
90031	SyncFromOtherMachine 不是 Number 类型或者其值范围不在 0~4294967295
91000	服务内部错误，请重试。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见[REST API调试工具](#)。

API 集成

PHP 集成

无。

参考

导入群消息 ([v4/group_open_http_svc/import_group_msg](#)) 。

消息推送 推送

最近更新时间：2018-08-14 15:27:12

功能说明

1. 支持全员推送；
2. 支持按用户属性推送；
3. 支持按用户标签推送；
4. 管理员向帐号推送消息，接收方看到消息发送者是管理员；
5. 管理员指定某一账户向其他账户推送消息，接收方看到发送者不是管理员，而是管理员指定的帐号；
6. 支持消息离线存储。
7. 目前推送接口可能存在延迟，因此暂不适用于对消息实时性要求较高的场景。

注意：当前推送功能比较适合低频次推送场景（例如每天或每周一次的营销推送）。如果推送频次过高，则推送可能会有延迟。

接口调用说明

本功能为增值服务，**需要申请开通才能使用**。

如有需要，请到云通信控制台相应 SDKAPPID "应用配置->功能配置->消息推送" 设置开启，收费方式请参考 [计费规则](#)；功能开启后，将在 50 分钟内生效。

请求 URL

```
https://console.tim.qq.com/v4/openim/im_push?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

本接口包含全员/属性/标签推送，总共最高每天 100 次，每两次推送间隔必须大于 1s。有特别频率需求的，请咨询客户经理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- 全员推送示例

管理员进行全员推送（发送方帐号为请求 URL 中的 identifier 参数，即管理员本身）：

```
{
  "MsgRandom": 20160201,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员指定某一帐号进行全员推送（例子中发送方帐号为xiaoming）：

```
{
  "MsgRandom": 20160204,
  "From_Account": "xiaoming",
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员给全员推送消息，并且消息离线保存 120 秒(即 2 分钟)：

```
{
  "MsgRandom": 20160204,
  "MsgLifeTime": 120,
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

- 按用户标签推送示例

管理员给带有关注“股票A”和“股票B”标签的用户推送消息：

```
{
  "MsgRandom": 20126201,
  "Condition": {
    "TagsAnd": ["股票A", "股票B"]
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员给带有关注“股票A”或“股票B”标签的用户推送消息：

```
{
  "MsgRandom": 20126201,
  "Condition": {
    "TagsOr": ["股票A", "股票B"]
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
```

```
"Text": "hi, beauty"
}
}
]
}
```

管理员给关注“股票A”或“股票B”的用户推送消息，并且消息离线保存 120 秒(即 2 分钟)：

```
{
  "MsgRandom": 20526201,
  "MsgLifeTime": 120,
  "Condition": {
    "TagsOr": ["股票A", "股票B"]
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

- **按用户属性推送**

管理员给在深圳的超白金会员用户推送消息：

```
{
  "MsgRandom": 20160204,
  "Condition": {
    "AttrsAnd": {
      "会员等级": "超白金会员",
      "city": "深圳"
    }
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员给在深圳或者性别为男的用户推送消息：

```
{
  "MsgRandom": 20160204,
  "Condition": {
    "AttrsOr": {
      "sex": "男",
      "city": "深圳"
    }
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

管理员给在深圳的超白金用户用户推送消息，并且消息离线保存 120 秒(即 2 分钟)：

```
{
  "MsgRandom": 2016204,
  "MsgLifeTime": 120,
  "Condition": {
    "AttrsAnd": {
      "会员等级": "超白金用户",
      "city": "深圳"
    }
  },
  "MsgBody": [
    {
      "MsgType": "TIMTextElem",
      "MsgContent": {
        "Text": "hi, beauty"
      }
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
----	----	----	----

字段	类型	属性	说明
Condition	Object	选填	Condition 包含 TagsOr 和 TagsAnd 两种属性条件类型。这两种可以类型可以并存。这时候表示同时满足 TagsOr 和 TagsAnd 的用户。如果没有 Condition，则推送给全部用户。
MsgRandom	Integer	必填	消息随机数,由随机函数产生。用于推送任务去重。对于不同的推送请求, MsgRandom7 天之内不能重复, 否则视为相同的推送任务(调用推送 API 返回失败的时候可以用相同的 MsgRandom 进行重试)
MsgBody	Object	必填	消息内容,具体格式请参考 MsgBody消息内容说明 。(一条消息可包括多种消息元素,所以MsgBody 为 Array 类型)
MsgType	String	必填	TIM 消息对象类型,目前支持的消息对象包括: TIMTextElem(文本消息), TIMFaceElem(表情消息), TIMLocationElem(位置消息), TIMCustomElem(自定义消息)。
MsgContent	Object	必填	对于每种 MsgType 用不同的 MsgContent 格式,具体可参考 TIMMsgElement 对象的定义。
MsgLifeTime	Integer	选填	消息离线存储时间,单位秒,最多保存1天(86400秒)。默认为0,表示不离线存储。
From_Account	String	选填	消息推送方帐号。
AttrsOr	Object	选填	属性条件的并集。注意属性推送和标签推送不可同时作为推送条件。
AttrsAnd	Object	选填	属性条件的交集。注意属性推送和标签推送不可同时作为推送条件。
TagsOr	Object	选填	标签条件的并集。标签是一个不超过 50 字节的字符串。注意属性推送和标签推送不可同时作为推送条件。TagsOr 条件中的标签个数不能超过 10 个
TagsAnd	Object	选填	标签条件的交集。标签是一个不超过 50 字节的字符串。注意属性推送和标签推送不可同时作为推送条件。TagsAnd 条件中的标签个数不能超过 10 个

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
```

```
"TaskId": "123456789_201621292_2016020204"
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
TaskId	String	推送任务 ID。可用于推送任务状态查询。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90002	Json 格式请求包中 MsgBody 不符合消息格式描述，或者 MsgBody 不是 Array 类型，请参考 MsgBody消息内容说明 。
90004	Json 格式请求包体中 MsgSeq 不是 Number 类型或者其值范围不在 0~4294967296。
90005	Json 格式请求包体中 MsgRandom 不是 Number 类型或者其值范围不在 0~4294967296。
90007	Json 格式请求包体中 MsgBody 类型不是 Array 类型，请将其修改为 Array 类型。
90009	非管理员账户，无权代发消息，请用管理员帐号使用该 Rest API 接口。
90010	Json 格式请求包不符合消息格式描述，请参考 TIMMsgElement对象 的定义。
90013	标签长度超过限制（不能超过 50 字节）。
90014	获取推送状态指定任务数过多。
90020	标签长度超过限制（不能超过 50 字节）。
90020	Tag 非法（必须是均为字符串的键值对，键的取值 "0"~"9"）。

错误码	含义说明
90024	推送频率超过限制。
90025	推送任务已存在（同一天内的不同推送请求 MsgRandom 必须不同，否则视为重复的推送任务）。
90026	消息离线存储时间错误（最多不能超过 1 天）。
90027	推送条件非法，请检查 Condition 字段。
91000	服务内部错误，请重试。
90032	TagsOr 或者 TagsAnd 中的标签个数不能超过 10 个
90039	按属性推送和按标签推送不可同时存在。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

获取推送报告

最近更新时间：2018-04-24 17:37:53

1. 功能说明

根据任务ID查询推送任务状态。注意：每次最多只能查询500个任务。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/openim/im_get_push_report?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{
  "TaskIds": [
    "144115197002484901_2016053001_2016053001",
    "144115197002484901_2016053001_2016053002"
  ]
}
```

2.7 请求包字段说明

字段	类型	属性	说明
TaskId	String	必填	推送任务ID。

2.8 应答包体示例

```

{
  "Reports": [
    {
      "TaskId": "123456789_201621292_2016020204",
      "Status": 0
    },
    {
      "TaskId": "123456789_201621292_2016020203",
      "Status": 2,
      "StartTime": "2016-02-02 17:02:19",
      "Finished": 900,
      "Total": 900
    }
  ],
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
    
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
Reports	Array	推送任务报告列表。
Status	Integer	0(未处理) / 1 (推送中) / 2 (推送完成)。注意：推送完成不代表完全推送成功，具体发送情况见Finished和Total字段数据。
StartTime	String	推送开始时间，Status不为0，才有这个字段。
Finished	Integer	已完成推送人数。
Total	Integer	需推送总人数。

2.10 错误码说明

除非发生网络错误(例如502错误), 该接口的HTTP返回码均为200。真正的错误码, 错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码 (60000到79999) 参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
90001	Json格式解析失败,请检查请求包是否符合JSON规范。
90009	非管理员不能查询
90014	推送任务数过多, 最多只能查询500个任务。
90015	缺少任务ID。
90016	推送任务ID非法, 请检查任务ID是否存在。
90017	查询失败, 请重试。
91000	服务内部错误, 请重试。
20002	签名验证失败。

3. 接口调试工具

Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4. 可能触发的回调

无。

5. 参考

无。

设置应用属性名称

最近更新时间：2018-04-24 17:39:04

功能说明

每个应用可以设置自定义的用户属性，最多可以有 10 个。通过本接口可以设置每个属性的名称，设置完成后，即可用于按用户属性推送等接口。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_set_attr_name?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

设置应用第0号属性表示性别，第1号属性表示城市，第2号属性表示国家。

```
{
  "AttrNames": {
    "0": "sex",
    "1": "city",
    "2": "country"
  }
}
```

```
}
}
```

请求包字段说明

字段	类型	属性	说明
数字键	String	必填	表示第几个属性，（“0”到“9”之间）。
属性名	String	必填	属性名最长不超过 50 字节。应用最多可以有 10 个推送属性（编号从 0 到 9），每个属性的含义用户自定义。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。

错误码	含义说明
90009	非管理员不能设置。
91000	服务内部错误，请重试。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

获取应用属性名称

最近更新时间：2018-04-24 17:42:36

功能说明

管理员获取应用属性名称。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_get_attr_name?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{}
```

请求包字段说明

无。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "AttrNames": {
    "0": "sex",
    "1": "city",
    "2": "会员等级"
  }
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
AttrNames	Object	包含多个键对。每对键值对，表示第几个属性对应的名称。比如 "0": "xxx" 表示第 0 号属性的名称是 xxx。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90009	非管理员不能设置。
90018	用户数量错误，不能超过 500 个。
90019	设置标签请求体非法。
90020	Tag 非法。
91000	服务内部错误，请重试。

错误码	含义说明
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

设置用户属性

最近更新时间：2018-04-24 17:44:43

功能说明

管理员给用户设置属性。每次最多只能给 500 个用户设置属性。注意使用前请先 [定义应用的用户属性](#)。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_set_attr?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "UserAttrs":
  [
    {
      "To_Account": "xiaojun012",
      "Attrs": {
        "sex": "attr1",
        "city": "attr2"
      }
    }
  ]
}
```

```

}
},
{
  "To_Account": "xiaojun013",
  "Attrs": {
    "city": "attr3",
    "sex": "attr4"
  }
}
]
}
    
```

请求包字段说明

字段	类型	属性	说明
To_Account	String	必填	目标用户帐号。
Attrs	Object	必填	属性集合。每个属性是一个键值对，键为属性名，值为该用户对应的属性值。用户属性值不能超过 50 字节。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误), 该接口的 HTTP 返回码均为 200。真正的错误码, 错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90009	非管理员不能设置。
90018	用户数量错误, 不能超过 500 个。
90019	属性请求体非法。
90020	属性非法。
90035	用户非法。
91000	服务内部错误, 请重试。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

删除用户属性

最近更新时间：2018-04-24 17:45:12

功能说明

管理员给用户删除属性。注意每次最多只能给 500 个用户删除属性。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/openim/im_remove_attr?usersig=xxx&identifier=admin&sdkappid=8888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "UserAttrs": [
    {
      "To_Account": "xiaojun013",
      "Attrs": [
        "sex",
        "city"
      ]
    }
  ]
}
```

```

},
{
  "To_Account": "xiaojun012",
  "Attrs": [
    "sex",
    "city"
  ]
}
]
}

```

请求包字段说明

字段	类型	属性	说明
To_Account	String	必填	目标用户帐号。
Attrs	Object	必填	属性集合，注意这里只需要给出属性名即可。Attrs 形式及含义参见 推送请求包字段说明 。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}

```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90009	非管理员不能删除属性。
90018	用户数量错误，不能超过 500 个。
90021	删除属性请求体非法。
90020	属性非法。
91000	服务内部错误，请重试。
90035	用户非法。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#)工具调试本接口。

可能触发的回调

无。

参考

无。

获取用户属性

最近更新时间：2018-04-24 17:45:46

功能说明

获取用户属性（必须以管理员帐号调用）。注意每次最多只能获取 500 个用户的属性。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_get_attr?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "To_Account": [
    "张小红",
    "陈小明",
    "abc"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
To_Account	Array	必填	目标用户帐号列表。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "UserAttrs": [
    {
      "To_Account": "张小红",
      "Attrs": {
        "sex": "女",
        "city": "纽约"
      }
    },
    {
      "To_Account": "abc",
      "Attrs": {}
    },
    {
      "To_Account": "陈小明",
      "Attrs": {
        "sex": "男",
        "city": "深圳"
      }
    }
  ]
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

字段	类型	说明
UserAttrs	Array	用户标签内容列表
To_Account	String	用户帐号
Attrs	Object	属性内容。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90018	用户数量错误，不能超过 500 个。
90023	获取属性请求体非法。
90035	用户非法。
91000	服务内部错误，请重试。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

添加用户标签

最近更新时间：2018-04-24 17:58:52

功能说明

管理员给用户添加标签。

注意：

1. 每次请求最多只能给 100 个用户添加标签，请求体中单个用户添加标签数最多为 10 个。
2. 单个用户可设置最大标签数为 100 个，若用户当前标签超过 100，则添加新标签之前请先删除旧标签。
3. 单个标签最大长度为 50 字节。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_add_tag?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```

{
  "UserTags": [
    {
      "To_Account": "xiaojun012",
      "Tags": ["a", "b"]
    },
    {
      "To_Account": "xiaojun013",
      "Tags": ["a", "b"]
    }
  ]
}
    
```

请求包字段说明

字段	类型	属性	说明
To_Account	String	必填	目标用户帐号。
Tags	Object	必填	标签集合。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误), 该接口的HTTP返回码均为 200。真正的错误码, 错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90009	非管理员不能设置。
90018	用户数量错误, 不能超过 500 个。
90019	添加标签请求体非法。
90020	Tag 非法。
90046	用户 Tag 超过最大个数限制。
91000	服务内部错误, 请重试。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

获取用户标签

最近更新时间：2017-03-09 10:31:08

1. 功能说明

获取用户标签（必须以管理员帐号调用）。注意每次最多只能获取500个用户的标签。

2 接口调用说明

2.1 请求URL

```
https://console.tim.qq.com/v4/openim/im_get_tag?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

2.2 请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

2.3 最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

2.4 HTTP请求方式

POST

2.5 HTTP请求包体格式

JSON

2.6 请求包示例

```
{
  "To_Account": [
    "xiaojun012",
    "xiaojun013"
  ]
}
```

2.7 请求包字段说明

字段	类型	属性	说明
To_Account	Array	必填	目标用户帐号列表。

2.8 应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "UserTags": [
    {
      "To_Account": "xiaojun012",
      "Tags": ["a", "b"]
    },
    {
      "To_Account": "xiaojun013",
      "Tags": ["a", "c"]
    }
  ]
}
    
```

2.9 应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
UserTags	Array	用户标签内容列表
To_Account	String	用户帐号
Tags	Object	Tags内容。Tag形式及含义参见 推送请求包字段说明 。

2.10 错误码说明

除非发生网络错误(例如502错误)，该接口的HTTP返回码均为200。真正的错误码，错误信息是通过应答包体中的ErrorCode、ErrorInfo来表示的。

公共错误码（60000到79999）参见[REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
90001	Json格式解析失败,请检查请求包是否符合JSON规范。
90009	非管理员不能删除标签。
90018	用户数量错误，不能超过500个。
90023	获取标签请求体非法。
91000	服务内部错误，请重试。
20002	签名验证失败。

3. 接口调试工具

Web调试工具

通过[REST API在线调试](#)工具调试本接口。

4. 可能触发的回调

无。

5. 参考

无。

删除用户标签

最近更新时间：2018-04-24 18:00:38

功能说明

管理员给用户删除标签。注意每次最多只能给500个用户删除标签。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_remove_tag?usersig=xxx&identifier=admin&sdkappid=8888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "UserTags": [
    {
      "To_Account": "xiaojun012",
      "Tags": ["a", "b"]
    },
    {
      "To_Account": "xiaojun013",
```

```
"Tags": ["a", "b"]
}
]
```

请求包字段说明

字段	类型	属性	说明
To_Account	String	必填	目标用户帐号。
Tags	Object	必填	标签集合。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。

错误码	含义说明
90009	非管理员不能删除标签。
90018	用户数量错误，不能超过 500 个。
90021	删除标签请求体非法。
90020	Tag 非法。
91000	服务内部错误，请重试。
20002	签名验证失败。

接口调试工具

Web 调试工具

通过 [REST API在线调试](#) 工具调试本接口。

可能触发的回调

无。

参考

无。

删除用户所有标签

最近更新时间：2018-04-24 18:01:38

功能说明

管理员给用户删除所有标签。注意每次最多只能给 500 个用户删除所有标签。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/im_remove_all_tags?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "To_Account": [
    "xiaojun012",
    "xiaojun013"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
To_Account	String	必填	目标用户帐号。
Tags	Object	必填	标签集合。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误(例如 502 错误)，该接口的 HTTP 返回码均为 200。真正的错误码，错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90009	非管理员不能删除标签。
90018	用户数量错误，不能超过 500 个。
91000	服务内部错误，请重试。
20002	签名验证失败。

接口调试工具

Web调试工具

通过 [REST API在线调试](#)工具调试本接口。

可能触发的回调

无。

参考

无。

计费规则

最近更新时间：2018-08-15 15:15:39

计费规则

全员推送服务计费：单次推送目标用户数小于 1 万的免费（每天可推送最多 100 次），超过限制即需要收费：

推送速度

超出限制默认使用共享通道：10万条/秒（共享），速度根据当前所有客户推送需求而定，不保证速度；

消息保存

离线保存消息条数：5 条

收费价格

默认100次/天（取当月峰值）；

101-500次/天，收费 2000元/月；

501-1000次/天，收费 4000元/月；

1001次及以上，在 4000元/月基础上、每1000次加收1000元/月

例：需求为280次/秒，则收费为 2000元/月；需求为 3500次/秒，则收费为 4000元/月+3*1000元/月=7000元/月

备注

功能开启后，将在 50 分钟内生效。

在线状态

获取用户在线状态

最近更新时间：2018-04-24 17:28:51

功能说明

获取用户当前的登录状态。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim/querystate?usersig=xxx&identifier=admin&sdkappid=8888888888&random=999999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "To_Account": ["id1", "id2", "id2"]
}
```

请求包字段说明

字段	类型	属性	说明
To_Account	Array	必填	需要查询这些 Identifier 的登录状态，一次最多查询 500 个 Identifier 的状态。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "QueryResult": [
    {
      "To_Account": "id1",
      "State": "Offline"
    },
    {
      "To_Account": "id2",
      "State": "Online"
    },
    {
      "To_Account": "id3",
      "State": "PushOnline"
    }
  ]
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
QueryResult	Array	返回的用户在线状态结构化信息。
To_Account	String	返回的用户的 Identifier。

字段	类型	说明
State	String	返回的用户状态，目前支持的状态有： 1."Online"：客户端登录后和云通信后台有长连接 2."PushOnline"：iOS 客户端退到后台或进程被杀或因网络问题掉线，进入 PushOnline 状态，此时仍然可以接收消息离线 APNS推送 。注意，云通信后台只会保存 PushOnline 状态 7 天时间，若从掉线时刻起 7 天之内未登录过，则进入 Offline 状态。 3."Offline"：客户端主动退出登录或者客户端自上一次登录起 7 天之内未登录过。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
90001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
90003	Json 格式请求包中 To_Account 不符合消息格式描述，请检查 To_Account 类型是否为 String。
90009	非管理员账户，请用管理员帐号使用该 Rest API 接口。
90011	请求消息中带的 To_Account 用户数目超过 500 个，请减少 To_Account 中帐号数量。
90992	后端服务超时，请重试。
91000	服务内部错误，请重试。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

可能触发的回调

无

参考

[用户在线状态综述](#)

资料管理

拉取资料

最近更新时间：2018-07-18 15:40:41

功能说明

1. 支持拉取好友和非好友的资料字段；
2. 支持拉取标配资料字段和自定义资料字段；
3. 建议每次拉取的用户数不超过100，避免因回包数据量太大导致回包失败。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/profile/portrait_get?usersig=xxx&identifier=admin&sdkappid=888888888&random=999999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- 读一个用户的一个资料字段

```
{
  "To_Account":["id1"],
  "TagList":
```

```
[
  "Tag_Profile_IM_Nick"
]
```

- 读一个用户的多个资料字段

```
{
  "To_Account":["id1"],
  "TagList":
  [
    "Tag_Profile_IM_Nick",
    "Tag_Profile_IM_AllowType",
    "Tag_Profile_IM_SelfSignature",
    "Tag_Profile_Custom_888888_Test"
  ]
}
```

- 读多个用户的一个资料字段

```
{
  "To_Account":["id1","id2","id3"],
  "TagList":
  [
    "Tag_Profile_IM_Nick"
  ]
}
```

- 读多个用户的多个资料字段

```
{
  "To_Account":["id1","id2","id3"],
  "TagList":
  [
    "Tag_Profile_IM_Nick",
    "Tag_Profile_IM_AllowType",
    "Tag_Profile_IM_SelfSignature",
    "Tag_Profile_Custom_888888_Test"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
----	----	----	----

字段	类型	属性	说明
To_Account	Array	必填	需要拉取这些 Identifier 的资料： 注意：每次拉取的用户数不得超过 100，避免因回包数据量太大以致回包失败。
TagList	Array	必填	指定要拉取的资料字段的 Tag，支持的字段有： 1、标配资料字段，详情可参见 标配资料字段 ； 2、自定义资料字段，详情可参见 自定义资料字段 。

应答

应答包体示例

- 读一个用户的一个资料字段

```
{
  "UserProfileItem":
  [
    {
      "To_Account":"id1",
      "ProfileItem":
      [
        {
          "Tag":"Tag_Profile_IM_Nick",
          "Value":"NickNameTest1"
        }
      ],
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"",
  "ErrorDisplay":""
}
```

- 读一个用户的多个资料字段

```
{
  "UserProfileItem":
  [
    {
```

```

    "To_Account": "id1",
    "ProfileItem":
    [
    {
    "Tag": "Tag_Profile_IM_Nick",
    "Value": "NickNameTest1"
    },
    {
    "Tag": "Tag_Profile_IM_AllowType",
    "Value": "AllowType_Type_NeedConfirm"
    },
    {
    "Tag": "Tag_Profile_IM_SelfSignature",
    "Value": "I'm Test1"
    },
    {
    "Tag": "Tag_Profile_Custom_888888_Test",
    "Value": "Custom Data1"
    }
    ],
    "ResultCode": 0,
    "ResultInfo": ""
    },
    "Fail_Account": [],
    "Invalid_Account": [],
    "ActionStatus": "OK",
    "ErrorCode": 0,
    "ErrorInfo": "",
    "ErrorDisplay": ""
    }
    
```

- 读多个用户的一个资料字段

```

    {
    "UserProfileItem":
    [
    {
    "To_Account": "id1",
    "ProfileItem":
    [
    {
    "Tag": "Tag_Profile_IM_Nick",
    "Value": "NickNameTest1"
    }
    ],
    }
    ],
    }
    
```

```

"ResultCode":0,
"ResultInfo":""
},
{
  "To_Account":"id2",
  "ProfileItem":
  [
    {
      "Tag":"Tag_Profile_IM_Nick",
      "Value":"NickNameTest2"
    }
  ],
  "ResultCode":0,
  "ResultInfo":""
},
{
  "To_Account":"id3",
  "ProfileItem":
  [
    {
      "Tag":"Tag_Profile_IM_Nick",
      "Value":"NickNameTest3"
    }
  ],
  "ResultCode":0,
  "ResultInfo":""
}
],
"Fail_Account":[],
"Invalid_Account":[],
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo":"","
"ErrorDisplay":""
}

```

- 读多个用户的多个资料字段

```

{
  "UserProfileItem":
  [
    {
      "To_Account":"id1",
      "ProfileItem":
      [
        {

```

```
"Tag":"Tag_Profile_IM_Nick",
"Value":"NickNameTest1"
},
{
"Tag":"Tag_Profile_IM_AllowType",
"Value":"AllowType_Type_NeedConfirm"
},
{
"Tag":"Tag_Profile_IM_SelfSignature",
"Value":"I'm Test1"
},
{
"Tag":"Tag_Profile_Custom_888888_Test",
"Value":"Custom Data1"
}
],
"ResultCode":0,
"ResultInfo":""
},
{
"To_Account":"id2",
"ProfileItem":
[
{
"Tag":"Tag_Profile_IM_Nick",
"Value":"NickNameTest2"
},
{
"Tag":"Tag_Profile_IM_AllowType",
"Value":"AllowType_Type_DenyAny"
},
{
"Tag":"Tag_Profile_IM_SelfSignature",
"Value":"I'm Test2"
},
{
"Tag":"Tag_Profile_Custom_888888_Test",
"Value":"Custom Data2"
}
],
"ResultCode":0,
"ResultInfo":""
},
{
"To_Account":"id3",
```

```

"ProfileItem":
[
{
"Tag":"Tag_Profile_IM_Nick",
"Value":"NickNameTest3"
},
{
"Tag":"Tag_Profile_IM_AllowType",
"Value":"AllowType_Type_AllowAny"
},
{
"Tag":"Tag_Profile_IM_SelfSignature",
"Value":"I'm Test3"
},
{
"Tag":"Tag_Profile_Custom_888888_Test",
"Value":"Custom Data3"
}
],
"ResultCode":0,
"ResultInfo":""
}
],
"Fail_Account":[],
"Invalid_Account":[],
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo":"","
"ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
UserProfileItem	Array	返回的用户资料结构化信息。
To_Account	String	返回的用户的Identifier。
ProfileItem	Array	返回的用户的资料对象数组，数组中每一个对象都包含了 Tag 和 Value。
Tag	String	返回的资料对象的名称： 1、标配资料字段，详情可参见 标配资料字段 ； 2、自定义资料字段，详情可参见 自定义资料字段 。

Value	uint64_t/string/bytes	拉取的资料对象的值，详情可参见 资料字段 。
ResultCode	Integer	返回的单个用户的结果，0 表示正确，非 0 表示错误。
ResultInfo	String	返回的单个用户的结果详细信息。
Fail_Account	Array	返回处理失败的用户列表。
Invalid_Account	Array	返回请求包中的非法用户列表。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
40001	资料系统解包失败： 如果是 REST API 返回的错误码，请对照 REST API 介绍文档仔细检查请求包格式； 如果是非 REST API 返回的错误，请联系技术客服。
40002	读请求没有指定 To_Account： 如果是 REST API 返回的错误码请对照 REST API 介绍文档仔细检查请求包格式； 如果是非 REST API 返回的错误请联系技术客服。
40003	帐号非法，一般是帐号不存在导致，请检查帐号的合法性或联系技术客服。
40004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
40007	读资料字段权限受限，请确认当前用户的读权限或联系技术客服。
40009	资料字段的 Tag 非法，请确认 Tag 的合法性： 1、标配资料字段的 Tag 可参见： 标配资料字段 ； 2、自定义资料字段的申请可参见： 自定义资料字段的申请 ； 3、如果资料字段的Tag正确无误（自定义资料字段需确保自定义资料字段的申请已生效），请联系技术客服。

40802	读标配资料字段的任务出错，请再次尝试或联系技术客服。
40803	解析标配资料字段失败，请再次尝试或联系技术客服。
40805	读自定义资料字段的任务出错，请再次尝试或联系技术客服。
40806	自定义资料字段的长度非法，请再次尝试或联系技术客服。
40808	读标配资料字段失败，请再次尝试或联系技术客服。
40810	读自定义资料字段失败，请再次尝试或联系技术客服。
40812	获取标配资料字段的缺省值失败，请再次尝试或联系技术客服。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#)中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取用户资料
./TimRestApiGear.php profile portrait_get (account_id)
```

更多调试工具可参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#)中，可以通过 `TimRestInterface` 的如下成员函数快速集成该 API：

```
// 获取用户资料
abstract function profile_portrait_get($account_id);

// 获取用户资料(高级接口)
abstract function profile_portrait_get2($account_list, $tag_list);
```

可能触发的回调

无

参考

REST API : [设置资料](#)

设置资料

最近更新时间：2018-07-18 15:40:58

功能说明

支持标配资料字段和自定义资料字段的设置。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/profile/portrait_set?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "ProfileItem":
  [
    {
      "Tag": "Tag_Profile_IM_Nick",
      "Value": "MyNickName"
    }
  ]
}
```

```

    ]
  }

```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要设置该Identifier的资料。
ProfileItem	Array	必填	待设置的用户的资料对象数组，数组中每一个对象都包含了 Tag 和 Value。
Tag	String	必填	指定要设置的资料字段的名称，支持设置的资料字段有： 1、标配资料字段，详情可参见 标配资料字段 ； 2、自定义资料字段，详情可参见 自定义资料字段 。
Value	uint64_t/string/bytes	必填	待设置的资料字段的值，详情可参见 资料字段

应答

应答包体示例

```

{
  "ActionStatus": "FAIL",
  "ErrorCode": 40001,
  "ErrorInfo": "Err_Profile_Comm_Decode_Fail",
  "ErrorDisplay": ""
}

```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
40001	资料系统解包失败： 如果是 REST API 返回的错误码，请对照 REST API 介绍文档仔细检查请求包格式； 如果是非 REST API 返回的错误，请联系技术客服。
40003	帐号非法，一般是帐号不存在导致，请检查帐号的合法性或联系技术客服。
40004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
40005	请求设置的资料字段中有敏感字，请检查资料字段或联系技术客服。
40008	写资料字段权限受限，请确认当前用户的写权限或联系技术客服。
40009	资料字段的 Tag 非法，请确认 Tag 的合法性： 1、标配资料字段的 Tag 可参见： 标配资料字段 ； 2、自定义资料字段的申请可参见： 自定义资料字段的申请 ； 3、如果资料字段的 Tag 正确无误（自定义资料字段需确保自定义资料字段的申请已生效），请联系技术客服。
40601	string/bytes 类型的资料字段的长度超门限，详情可参见 资料字段 。
40602	读标配资料失败，请再次尝试或联系技术客服。
40605	标配资料字段的值非法，标配资料字段的合理取值范围可参见： 标配资料字段 。
40606	写标配资料失败，请再次尝试或联系技术客服。
40608	自定义资料字段的值类型非法，请联系技术客服。
40609	写自定义资料失败，请再次尝试或联系技术客服。
40610	请求设置的资料字段的值类型非法，请确认值类型的合法性或联系技术客服。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#)调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#)中的调试工具进行简单的 REST API 调试，命令如下：

```
# 修改用户昵称
./TimRestApiGear.php profile portrait_set (account_id) (new_name)
```

更多调试工具参见[REST API调试工具](#)。

API 集成

PHP集成

在云通信[PHP Server SDK](#)中，可以通过TimRestInterface的如下成员函数快速集成该API：

```
// 修改用户昵称
abstract function profile_portrait_set($account_id, $new_name);

// 修改用户资料 (高级接口)
abstract function profile_portrait_set2($account_id, $profile_list);
```

可能触发的回调

无。

参考

REST API：[拉取资料](#)

关系链管理

添加好友

最近更新时间：2018-04-19 18:23:33

功能说明

支持批量添加好友。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_add?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[REST API调用频率调整](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

基础形式

```
{
  "From_Account": "id",
  "AddFriendItem":
  [
```

```
{
  "To_Account":"id1",
  "AddSource":"AddSource_Type_XXXXXXXX"
}
]
```

完整形式

```
{
  "From_Account":"id",
  "AddFriendItem":
  [
    {
      "To_Account":"id1",
      "Remark":"remark1",
      "GroupName":"同学",
      "AddSource":"AddSource_Type_XXXXXXXX",
      "AddWording":"I'm Test1"
    }
  ],
  "AddType":"Add_Type_Both",
  "ForceAddFlags":1
}
```

批量加好友

```
{
  "From_Account":"id",
  "AddFriendItem":
  [
    {
      "To_Account":"id1",
      "AddSource":"AddSource_Type_XXXXXXXX"
    },
    {
      "To_Account":"id2",
      "Remark":"remark2",
      "GroupName":"同学",
      "AddSource":"AddSource_Type_XXXXXXXX",
      "AddWording":"I'm Test2"
    },
    {
      "To_Account":"id3",
      "Remark":"remark3",

```

```

"GroupName": "同事",
"AddSource": "AddSource_Type_XXXXXXX",
"AddWording": "I'm Test3"
}
],
"AddType": "Add_Type_Both",
"ForceAddFlags": 1
}
    
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要为该 Identifier 添加好友。
AddFriendItem	Array	必填	好友结构体对象。
To_Account	String	必填	好友的 Identifier。
Remark	String	选填	From_Account 对 To_Account 的好友备注，详情可参见 标配好友字段 。
GroupName	String	选填	From_Account 对 To_Account 的分组信息，详情可参见 标配好友字段 。
AddSource	String	必填	加好友来源字段，详情可参见 标配好友字段 。
AddWording	String	选填	From_Account 和 To_Account 形成好友关系时的附言信息，详情可参见 标配好友字段 。
AddType	String	选填	加好友方式（默认双向加好友方式）： "Add_Type_Single" 表示单向加好友； "Add_Type_Both" 表示双向加好友。
ForceAddFlags	Integer	选填	管理员强制加好友标记：1 表示强制加好友；0 表示常规加好友方式。

应答

应答包体示例

基础形式、完备形式

```

{
  "ResultItem":
  [
    
```

```

{
  "To_Account":"id1",
  "ResultCode":0,
  "ResultInfo":""
}
],
"Fail_Account":[],
"Invalid_Account":[],
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo":"",
"ErrorDisplay":""
}
    
```

批量加好友

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id2",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id3",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"",
  "ErrorDisplay":""
}
    
```

应答包字段说明

--	--	--

字段	类型	说明
ResultItem	Array	批量加好友的结果对象数组。
To_Account	String	请求添加的好友的 Identifier。
ResultCode	Integer	批量加好友中单个好友的处理结果，0 表示业务成功，非 0 表示业务失败。
Fail_Account	Array	返回处理失败的用户列表。
Invalid_Account	Array	返回请求包中的非法用户列表。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段或联系技术客服。
30501	加好友来源字段非法，详情可参见 标配好友字段 。
30502	请求包中字符串类型的参数长度非法，详情可参见 标配好友字段 ，可能的错误有： 1、好友备注的长度非法； 2、分组名称的长度非法； 3、加好友附言的长度非法。

30503	请求添加的好友个数非法，好友个数的合理取值范围是[1, 1000]。
30504	加好友请求获取 SDKAppId 失败，请再次尝试或联系技术客服。
30505	加好友请求的 Task 出错，请再次尝试或联系技术客服。
30506	获取 From_Account 的好友列表失败，请再次尝试或联系技术客服。
30507	获取 From_Account 的好友列表元数据失败，请再次尝试或联系技术客服。
30508	拉取 From_Account 的好友列表行数据失败，请再次尝试或联系技术客服。
30509	更新 From_Account 的好友列表行数据错误，请再次尝试或联系技术客服。
30510	更新 From_Account 的好友列表元数据错误，请再次尝试或联系技术客服。
30511	更新 From_Account 的好友信息错误，请再次尝试或联系技术客服。
30512	删除 From_Account 的未决信息错误，请再次尝试或联系技术客服。
30513	拉取 From_Account 的黑名单列表错误，请再次尝试或联系技术客服。
30514	添加好友新建 Task 失败，请再次尝试或联系技术客服。
30515	To_Account 在 From_Account 的黑名单列表中，无法添加好友，调用方可以捕捉该错误给用户一个合理的提示。
30516	To_Account 的加好友验证方式的是“不允许任何人添加”，无法添加好友，调用方可以捕捉该错误给用户一个合理的提示。
30517	本次请求有超过 100 个帐号设置的加好友验证方式是“需要通过验证才允许添加”，调用方可以捕捉该错误给用户一个合理的提示。
30518	分组个数达到系统上限：请求添加的好友携带了新的分组信息，但是已有的分组个数加上新的分组的个数超过了 32 个，调用方可以捕捉该错误给用户一个合理的提示。
30519	好友表人数（包含本次请求中可直接添加为好友的人数）已达上限，调用方可以捕捉该错误给用户一个合理的提示。
30520	From_Account 与 To_Account 之间已经存在好友关系，调用方可以捕捉该错误给用户一个合理的提示。
30521	To_Account 在资料系统中没有任何数据，调用方可以捕捉该错误给用户一个合理的提示。
30522	获取 To_Account 的资料信息错误，请再次尝试或联系技术客服。
30523	获取 To_Account 的资料信息行数据错误，请再次尝试或联系技术客服。
30524	获取 To_Account 的黑名单信息错误，请再次尝试或联系技术客服。

30525	From_Account 在 To_Account 的黑名单中，无法添加为好友，调用方可以捕捉该错误给用户一个合理的提示。
30526	获取 To_Account 的关系链数据错误，请再次尝试或联系技术客服。
30527	获取 To_Account 的关系链数据时解行数据错误，请再次尝试或联系技术客服。
30528	获取 To_Account 的未决 sequence 错误，请再次尝试或联系技术客服。
30529	获取 To_Account 的未决 sequence 时解元数据错误，请再次尝试或联系技术客服。
30530	获取 To_Account 的未决数据错误，请再次尝试或联系技术客服。
30531	获取 To_Account 的未决数据时解行数据错误，请再次尝试或联系技术客服。
30532	删除 To_Account 的未决数据失败，请再次尝试或联系技术客服。
30533	获取 To_Account 的好友个数失败，请再次尝试或联系技术客服。
30534	获取 To_Account 的好友个数时序列号不一致，请再次尝试或联系技术客服。
30535	To_Account 的好友个数已达上限，调用方可以捕捉该错误给用户一个合理的提示。
30536	更新 To_Account 的好友表时封装元数据错误，请再次尝试或联系技术客服。
30537	更新 To_Account 的好友表时封装行数据错误，请再次尝试或联系技术客服。
30538	更新 To_Account 的好友表错误，请再次尝试或联系技术客服。
30539	A发起加好友请求，请求添加 B 为好友，B 的加好友验证方式被设置为“AllowType_Type_NeedConfirm”，这时 A与 B 之间只能形成未决关系，这个返回码用于标识加未决成功，以便与加好友成功的返回码区分开，调用方可以捕捉该错误给用户一个合理的提示。
30540	加好友请求被安全策略打击，请勿频繁发起加好友请求。
30541	加好友请求的加好友类型非法，加好友类型的合理取值范围是：“Add_Type_Single”，“Add_Type_Both”。
30542	请求强制加好友，但不具备管理员权限，请确认权限或联系技术客服。
30543	写冲突导致更新 From_Account 的未决列表失败。
30544	写冲突导致更新 From_Account 的好友列表失败。
30545	后台禁止 From_Account 发起加好友请求。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

API 集成

无。

可能触发的回调

添加好友可能触发的回调有：

- 1、[添加好友之前回调](#)；
- 2、[添加好友之后回调](#)。

参考

REST API：[导入好友](#)

REST API：[更新好友](#)

REST API：[删除好友](#)

REST API：[删除所有好友](#)

REST API：[校验好友](#)

REST API：[拉取好友](#)

REST API：[拉取指定好友](#)

导入好友

最近更新时间：2018-04-19 18:24:49

功能说明

1. 支持批量导入单向好友；
2. 往同一个用户导入好友时建议采用批量导入的方式，避免并发写导致的写冲突。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_import?usersig=xxx&identifier=admin&sdkappid=888888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

基础形式

```
{
  "From_Account": "id",
  "AddFriendItem":
  [
    {
```

```
"To_Account":"id1",
"AddSource":"AddSource_Type_XXXXXXXX"
}
]
}
```

完整形式

```
{
"From_Account":"id",
"AddFriendItem":
[
{
"To_Account":"id1",
"Remark":"remark1",
"RemarkTime":1420000001,
"GroupName":["朋友"],
"AddSource":"AddSource_Type_XXXXXXXX",
"AddWording":"I'm Test1",
"AddTime":1420000001,
"CustomItem":
[
{
"Tag":"Tag_SNS_1300000000_XXXX",
"Value":"Test"
},
{
"Tag":"Tag_SNS_1300000000_YYYY",
"Value":0
}
]
}
]
}
```

批量加好友

```
{
"From_Account":"id",
"AddFriendItem":
[
{
"To_Account":"id1",
"AddSource":"AddSource_Type_XXXXXXXX"
},

```

```

{
  "To_Account":"id2",
  "Remark":"remark2",
  "RemarkTime":1420000001,
  "GroupName":["朋友"],
  "AddSource":"AddSource_Type_XXXXXXXX",
  "AddWording":"I'm Test2",
  "AddTime":1420000001
},
{
  "To_Account":"id3",
  "Remark":"remark3",
  "RemarkTime":1420000001,
  "GroupName":["同事","朋友"],
  "AddSource":"AddSource_Type_XXXXXXXX",
  "AddWording":"I'm Test3",
  "AddTime":1420000001,
  "CustomItem":
  [
    {
      "Tag":"Tag_SNS_1300000000_XXXX",
      "Value":"Test"
    },
    {
      "Tag":"Tag_SNS_1300000000_YYYY",
      "Value":0
    }
  ]
}
    
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要为该 Identifier 添加好友。
AddFriendItem	Array	必填	好友结构体对象。
To_Account	String	必填	好友的 Identifier。
Remark	String	选填	From_Account 对 To_Account 的好友备注，详情可参见 标 配好友字段 。

字段	类型	属性	说明
RemarkTime	Integer	选填	From_Account 对 To_Account 的好友备注时间。
GroupName	Array	选填	From_Account 对 To_Account 的分组信息，详情可参见 标配好友字段 。
AddSource	String	必填	加好友来源字段，详情可参见 标配好友字段 。
AddWording	String	选填	From_Account 和 To_Account 形成好友关系时的附言信息，详情可参见 标配好友字段 。
AddTime	Integer	选填	From_Account 和 To_Account 形成好友关系的时间。
CustomItem	Array	选填	From_Account 对 To_Account 的自定义表示对象信息数组，每一个对象都包含了 Tag 和 Value。
Tag	String	选填	From_Account 对 To_Account 的自定义标签，使用前需要联系云通讯团队申请一个 Tag。
Value	String/Integer	选填	From_Account 对 To_Account 的自定义内容，可以是字符串，也可以是一个整数或者是一段 Buffer。

应答

应答包体示例

基础形式、完备形式

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"","
  "ErrorDisplay":""
}
    
```

批量加好友

```

{
  "ResultItem":
  [
    {
      "To_Account": "id1",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id2",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id3",
      "ResultCode": 0,
      "ResultInfo": ""
    }
  ],
  "Fail_Account": [],
  "Invalid_Account": [],
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "",
  "ErrorDisplay": ""
}
    
```

应答包字段说明

字段	类型	说明
ResultItem	Array	批量加好友的结果对象数组。
To_Account	String	请求添加的好友的 Identifier。
ResultCode	Integer	批量加好友中单个好友的处理结果，0 表示业务成功，非 0 表示业务失败。
Fail_Account	Array	返回处理失败的用户列表。
Invalid_Account	Array	返回请求包中的非法用户列表。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。

字段	类型	说明
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整行； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段或联系技术客服。
30801	请求包参数错误，请检查请求包的合法性，详情可参见 标配好友字段 。
30802	好友导入获取 SDKAppId 失败，请再次尝试或联系技术客服。
30803	好友导入读好友数错误，请再次尝试或联系技术客服。
30804	好友导入获取组信息和序列号错误，请再次尝试或联系技术客服。
30805	好友导入获取组信息和序列号元数据错误，请再次尝试或联系技术客服。
30806	好友导入封装标配信息行数据错误，请再次尝试或联系技术客服。
30807	好友导入封装标配信息元数据错误，请再次尝试或联系技术客服。
30808	好友导入更新标配信息错误，请再次尝试或联系技术客服。
30809	好友导入封装自定义信息行数据错误，请再次尝试或联系技术客服。

错误码	详细错误信息
30810	好友导入封装自定义信息元数据错误，请再次尝试或联系技术客服。
30811	好友导入更新自定义信息错误，请再次尝试或联系技术客服。
30812	好友导入好友个数已达上限，调用方可以捕捉该错误给用户一个合理的提示。
30813	好友导入好友分组个数已达上限，调用方可以捕捉该错误给用户一个合理的提示。
30814	向同一个用户并发导入好友导致写冲突，建议改成批量导入方式，避免并发写同一个用户的数据。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 建立好友关系
./TimRestApiGear.php sns friend_import (account_id) (receiver)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 `TimRestInterface` 的如下成员函数快速集成该 API：

```
// 建立好友关系
abstract function sns_friend_import($accout_id, $receiver);
```

可能触发的回调

无

参考

REST API : [添加好友](#)

REST API : [更新好友](#)

REST API : [删除好友](#)

REST API : [删除所有好友](#)

REST API : [校验好友](#)

REST API : [拉取好友](#)

REST API : [拉取指定好友](#)

更新好友

最近更新时间：2018-04-19 18:25:04

功能说明

1. 支持批量更新同一用户的多个好友的关系链数据；
2. 更新一个用户多个好友时，建议采用批量方式，避免并发写导致的写冲突。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_update?usersig=xxx&identifier=admin&sdkappid=888888888&random=999999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100 次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

基础形式

```
{
  "From_Account": "id",
  "UpdateItem":
  [
    {
```

```

    "To_Account":"id1",
    "SnsItem":
    [
    {
    "Tag":"Tag_SNS_IM_Remark",
    "Value":"remark1"
    }
    ]
    }
    ]
    }
    
```

完整形式

```

{
  "From_Account":"id",
  "UpdateItem":
  [
  {
    "To_Account":"id1",
    "SnsItem":
    [
    {
      "Tag":"Tag_SNS_IM_Remark",
      "Value":"remark1"
    },
    {
      "Tag":"Tag_SNS_IM_Group",
      "Value":"group1"
    },
    {
      "Tag":"Tag_SNS_Custom_Test",
      "Value":"test"
    }
    ]
  }
  ]
}
    
```

批量加好友

```

{
  "From_Account":"id",
  "UpdateItem":
  [
    
```

```
{
  "To_Account":"id1",
  "SnsItem":
  [
    {
      "Tag":"Tag_SNS_IM_Remark",
      "Value":"remark1"
    }
  ]
},
{
  "To_Account":"id2",
  "SnsItem":
  [
    {
      "Tag":"Tag_SNS_IM_Remark",
      "Value":"remark2"
    },
    {
      "Tag":"Tag_SNS_IM_Group",
      "Value":"group2"
    }
  ]
},
{
  "To_Account":"id3",
  "SnsItem":
  [
    {
      "Tag":"Tag_SNS_IM_Remark",
      "Value":"remark3"
    },
    {
      "Tag":"Tag_SNS_IM_Group",
      "Value":"group3"
    },
    {
      "Tag":"Tag_SNS_Custom_Test",
      "Value":"test"
    }
  ]
}
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要更新该 Identifier 的关系链数据。
UpdateItem	Array	必填	需要更新的好友对象数组。
To_Account	String	必填	好友的 Identifier。
SnsItem	Array	必填	需要更新的关系链数据对象数组
Tag	String	必填	需要更新的关系链字段的字段名，目前只支持好友备注、好友分组、关系链自定义字段的更新操作，关系链字段的详细信息可参见 好友表 。
Value	Array/String/Integer	必填	需要更新的关系链字段的值，关系链字段的值类型信息可参见 好友表 。

应答

应答包体示例

基础形式、完备形式

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"","
  "ErrorDisplay":""
}
    
```

批量加好友

```

{
  "ResultItem":
    
```

```
[
  {
    "To_Account":"id1",
    "ResultCode":0,
    "ResultInfo":""
  },
  {
    "To_Account":"id2",
    "ResultCode":0,
    "ResultInfo":""
  },
  {
    "To_Account":"id3",
    "ResultCode":0,
    "ResultInfo":""
  }
],
"Fail_Account":[],
"Invalid_Account":[],
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo":"",
"ErrorDisplay":""
}
```

应答包字段说明

字段	类型	说明
ResultItem	Array	批量更新好友的结果对象数组。
To_Account	String	请求更新的好友的 Identifier。
ResultCode	Integer	批量更新好友的单个好友的处理结果，0 表示业务成功，非 0 表示业务失败。
Fail_Account	Array	返回处理失败的好友账号列表。
Invalid_Account	Array	返回请求包中的非法账号列表。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段的值的合法性或联系技术客服。
30701	好友数据更新参数错误，请检查备注内容的长度是否超过 96 字节，分组名称的长度是否超过 30 字节，多个分组名称出现是否为空，Tag 名称是否非法，目前的 Tag 名称只支持：“Tag_SNS_IM_Remark”和“Tag_SNS_IM_Group”。
30702	好友数据更新获取好友失败，请再次尝试或联系技术客服。
30703	好友数据更新获取好友元数据失败，请再次尝试或联系技术客服。
30704	好友数据更新获取好友行数据失败，请再次尝试或联系技术客服。
30705	好友数据更新标配信息行数据失败，请再次尝试或联系技术客服。
30706	好友数据更新标配信息元数据失败，请再次尝试或联系技术客服。
30707	好友数据更新标配信息失败，请再次尝试或联系技术客服。
30708	好友数据更新自定义信息行数据失败，请再次尝试或联系技术客服。
30709	好友数据更新自定义信息元失败，请再次尝试或联系技术客服。
30710	好友数据更新自定义信息失败，请再次尝试或联系技术客服。
30711	好友数据更新执行任务失败，请再次尝试或联系技术客服。
30712	好友数据更新的好友不存在，在自己的好友表中没有找到需要更新的好友，调用方可以捕捉该错误给用户一个合理的提示。

错误码	详细错误信息
30713	好友数据更新分组个数已达系统上限，详情可参见标配好友字段，调用方可以捕捉该错误给用户一个合理的提示。

接口调试工具

通过[REST API调试工具](#)调试本接口。

API 集成

无。

可能触发的回调

无。

参考

REST API：[添加好友](#)

REST API：[导入好友](#)

REST API：[删除好友](#)

REST API：[删除所有好友](#)

REST API：[校验好友](#)

REST API：[拉取好友](#)

REST API：[拉取指定好友](#)

删除好友

最近更新时间：2018-08-30 11:05:57

功能说明

支持清空所有好友。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_delete?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

单向删除好友

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3"],
  "DeleteType": "Delete_Type_Single"
}
```

双向删除好友

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3"],
  "DeleteType": "Delete_Type_Both"
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要删除该 Identifier 的好友。
To_Account	String	必填	待删除的好友的 Identifier 列表，单次请求的 To_Account 数不得超过 1000。
DeleteType	String	选填	删除模式，详情可参见 删除好友 。

应答

应答包体示例

```
{
  "ResultItem":
  [
    {
      "To_Account": "id1",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id2",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id3",
      "ResultCode": 0,
      "ResultInfo": ""
    }
  ],
  "Fail_Account": [],
  "Invalid_Account": [],
  "ActionStatus": "OK",
  "ErrorCode": 0,
}
```

```
"ErrorInfo": "0",
"ErrorDisplay": ""
}
```

应答包字段说明

字段	类型	说明
ResultItem	Array	批量删除好友的结果对象数组。
To_Account	String	请求删除的好友的 Identifier。
ResultCode	Integer	To_Account 的处理结果，0 表示删除成功，非 0 表示删除失败。
ResultInfo	String	To_Account 的处理详情。
Fail_Account	Array	返回处理失败的 To_Account 列表。
Invalid_Account	Array	返回请求包中的非法 To_Account 列表。
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。

错误码	详细错误信息
31701	请求删除的 To_Account 个数非法，详情可参见 请求包字段说明 。
31702	请求包中的删除模式字段非法，详情可参见 请求包字段说明 。
31703	校验正向好友关系失败，请再次尝试或联系技术客服。
31704	To_Account 与 From_Account 之间不存在好友关系，调用方可以捕捉该错误码给用户一个合理的提示。
31705	解除正向好友关系时清除标配关系链数据失败，请再次尝试或联系技术客服。
31706	解除反向好友关系失败，请再次尝试或联系技术客服。
31707	删除频率过高，该请求被判定为恶意请求，请暂停好友删除操作。
31708	并发删除好友导致写冲突，建议使用批量删除接口。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 解除双向好友关系
./TimRestApiGear.php sns friend_delete (account_id) (frd_id)
```

更多调试工具参见 [REST API调试工具](#)。

API集成

PHP集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 解除双向好友关系
abstract function sns_friend_delete($account_id, $frd_id);
```

可能触发的回调

参见 [删除好友之后回调](#)。

参考

REST API : [添加好友](#)

REST API : [导入好友](#)

REST API : [更新好友](#)

REST API : [删除所有好友](#)

REST API : [校验好友](#)

REST API : [拉取好友](#)

REST API : [拉取指定好友](#)

删除所有好友

最近更新时间：2018-04-19 18:25:27

功能说明

支持清空所有好友。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/sns/friend_delete_all?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{  
  "From_Account": "id",  
}
```

请求包字段说明

字段	类型	属性	说明
----	----	----	----

字段	类型	属性	说明
From_Account	String	必填	需要删除该 Identifier 的所有好友。

应答

应答包体示例

```
{
  "ActionStatus":"FAIL",
  "ErrorCode":30003,
  "ErrorInfo":"Err_Sns_Invalid_Account",
  "ErrorDisplay":""
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。

错误码	详细错误信息
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
32901	校验 SDKAppId 失败，请再次尝试或联系技术客服。
32902	删除标配数据失败，请再次尝试或联系技术客服。
32903	删除自定义数据失败，请再次尝试或联系技术客服。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 删除某一用户的所有好友
./TimRestApiGear.php sns friend_delete_all (account_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 删除某一用户的所有好友
abstract function sns_friend_delete_all($account_id);
```

可能触发的回调

无

参考

REST API : [添加好友](#)

REST API : [导入好友](#)

REST API : [更新好友](#)

REST API : [删除好友](#)

REST API : [校验好友](#)

REST API : [拉取好友](#)

REST API : [拉取指定好友](#)

校验好友

最近更新时间：2018-04-19 18:25:44

功能说明

支持批量校验好友关系。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_check?usersig=xxx&identifier=admin&sdkappid=888888888&random=999999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[REST API调用频率调整](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3", "id4"],
  "CheckType": "CheckResult_Type_Singal"
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要校验该 Identifier 的好友。
To_Account	Array	必填	请求校验的好友的 Identifier 列表，单次请求的 To_Account 数不得超过 1000。
CheckType	String	必填	校验模式，详情可参见 校验好友 。

应答

应答包体示例

```

{
  "CheckItem":
  [
    {
      "To_Account":"id1",
      "Relation":"CheckResult_Type_NoRelation",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id2",
      "Relation":"CheckResult_Type_AWithB",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":["id3"],
  "Invalid_Account":["id4"],
  "ActionStatus":"FAIL",
  "ErrorCode":30003,
  "ErrorInfo":"Err_Sns_Invalid_Account",
  "ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
CheckItem	Array	校验结果对象数组。
To_Account	String	请求校验的用户的 Identifier。

字段	类型	说明
Relation	String	校验成功时 To_Account 与 From_Account 之间的好友关系，详情可参见 校验好友
ResultCode	Integer	To_Account 的处理结果，0 表示校验成功，非0表示校验失败。
ResultInfo	String	To_Account 的处理详情。
Fail_Account	Array	返回处理失败的 To_Account 列表。
Invalid_Account	Array	返回请求包中的非法 To_Account 列表。
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是REST API请求，请参照REST API文档确认请求包的合法性和完整性； 如果不是REST API请求，请联系技术客服。
30002	SDKAppId非法，请检查SDKAppId及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自REST API的请求，但是不具备管理员权限，请联系技术客服。
30901	校验类型非法或请求校验的To_Account个数非法，详情可参见 请求包字段说明
30902	校验反向好友关系时任务执行失败，请再次尝试或联系技术客服。
30903	拉正向好友数据失败，请再次尝试或联系技术客服。
30904	拉正向好友关系时解行数据失败，请再次尝试或联系技术客服。

错误码	详细错误信息
30905	校验反向好友关系时新建任务失败，请再次尝试或联系技术客服。
30906	拉反向好友数据失败，请再次尝试或联系技术客服。
30907	拉反向好友数据时解行数据失败，请再次尝试或联系技术客服。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 验证双向好友关系
./TimRestApiGear.php sns friend_check (account_id) (to_account)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过TimRestInterface的如下成员函数快速集成该API：

```
// 校验双向好友关系
abstract function sns_friend_check($account_id, $to_account);

// 校验好友关系（高级接口）
abstract function sns_friend_check2($account_id, $to_account_list, $check_type);
```

可能触发的回调

无。

参考

REST API : [添加好友](#)

REST API : [导入好友](#)

REST API : [更新好友](#)

REST API : [删除好友](#)

REST API : [删除所有好友](#)

REST API : [拉取好友](#)

REST API : [拉取指定好友](#)

拉取好友

最近更新时间：2018-06-26 10:14:37

功能说明

支持分页拉取所有好友。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/friend_get_all?usersig=xxx&identifier=admin&sdkappid=888888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100 次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "TimeStamp": 0,
  "StartIndex": 0,
  "TagList":
  [
    "Tag_Profile_IM_Nick",
    "Tag_SNS_IM_Group",
  ]
}
```

```

"Tag_SNS_IM_Remark"
],
"LastStandardSequence":0,
"GetCount":100
}
    
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要拉取该 Identifier 的好友。
TimeStamp	Integer	选填	上次拉取的时间戳，不填或为 0 时表示全量拉取。
StartIndex	Integer	必填	拉取的起始位置。
TagList	Array	选填	指定要拉取的字段 Tag，支持拉取的字段有： 1、标配资料字段，详情可参见 标配资料字段 ； 2、自定义资料字段，详情可参见 自定义资料字段 ； 3、标配好友字段，详情可参见 标配好友字段 ； 4、自定义好友字段，详情可参见 自定义好友字段 ；
LastStandardSequence	Integer	选填	上次拉取标配关系链的 Sequence，仅在只拉取标配关系链字段时有用。
GetCount	Integer	选填	每页需要拉取的好友数量： 1、默认每页返回 100 个好友； 2、每页最多返回 100 个好友的数据； 3、如果拉取好友超时，请适量减少每页拉取的好友数。

应答

应答包体示例

```

{
  "NeedUpdateAll":"GetAll_Type_Yes",
  "TimeStampNow":1462334521,
  "StartIndex":0,
  "Infoltem":
  [
    {
      "Info_Account":"id1",
      "SnsProfileItem":
      [
        {
    
```

```

"Tag":"Tag_Profile_IM_Nick",
"Value":"NickTest1"
},
{
"Tag":"Tag_SNS_IM_Group",
"Value":["Group1"]
},
{
"Tag":"Tag_SNS_IM_Remark",
"Value":"Remark1"
}
]
},
{
"Info_Account":"id2",
"SnsProfileItem":
[
{
"Tag":"Tag_Profile_IM_Nick",
"Value":"NickTest2"
},
{
"Tag":"Tag_SNS_IM_Group",
"Value":["Group1","Group2"]
},
{
"Tag":"Tag_SNS_IM_Remark",
"Value":"Remark2"
}
]
}
],
"CurrentStandardSequence": 2,
"FriendNum": 2,
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo":"",
"ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
NeedUpdateAll	String	是否需要全量更新： "GetAll_Type_YES" 表示需要全量更新； "GetAll_Type_NO"表示不需要全量更新。
TimeStampNow	Integer	本次拉取的时间戳，客户端需要保存该时间，下次请求时通过 TimeStamp 字段返回给后台。
StartIndex	Integer	下页拉取的起始位置。
InfoItem	Array	好友对象数组，每一个好友对象都包括了 Info_Account 和 SnsProfileItem。
Info_Account	String	好友的 Identifier。
SnsProfileItem	Array	好友的详细信息数组，数组每一个元素都包括 Tag 和 Value。
Tag	String	好友的资料字段或好友字段的名称。
Value	String/Integer/Array	好友的资料字段或好友字段的值，详情可参见 关系链字段 及 资料字段 ；
CurrentStandardSequence	Integer	本次拉取标配关系链的 Sequence，客户端需要保存该 Sequence，下次请求时通过 LastStandardSequence 字段返回给后台。
FriendNum	Integer	好友总数。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
31101	拉取所有好友参数错误，请检查参数的合法性。
31102	拉取所有好友时字段数目异常，请求包体中的字段数量超过了 30 个。
31103	请求拉取好友的用户没有关系链标配数据，请确认当前用户是否有关系链标配数据。
31104	拉取所有好友获取序列号错误，请再次尝试或联系技术客服。
31105	拉取所有好友获取序列号元数据错误，请再次尝试或联系技术客服。
31106	拉取所有好友获取字段对应的字符串错误，请再次尝试或联系技术客服。
31107	拉取所有好友关系链标配信息错误，请再次尝试或联系技术客服。
31108	拉取所有好友关系链标配信息组数据错误，请再次尝试或联系技术客服。
31109	拉取所有好友关系链标配信息行数据错误，请再次尝试或联系技术客服。
31110	拉取所有好友关系链自定义信息错误，请再次尝试或联系技术客服。
31111	拉取所有好友关系链自定义信息行数据错误，请再次尝试或联系技术客服。
31112	拉取所有好友执行任务失败，请再次尝试或联系技术客服。

接口调试工具

Web 调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取用户所有好友  
./TimRestApiGear.php sns friend_get_all (account_id)
```

更多调试工具参见 [REST API调试工具](#)。

API集成

PHP集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该API：

```
// 拉取某一用户的所有好友  
abstract function sns_friend_get_all($account_id);  
  
// 拉取某一用户的所有好友（高级接口）  
abstract function sns_friend_get_all2($account_id, $tag_list);
```

可能触发的回调

无。

参考

REST API：[添加好友](#)

REST API：[导入好友](#)

REST API：[更新好友](#)

REST API：[删除好友](#)

REST API：[删除所有好友](#)

REST API：[校验好友](#)

REST API：[拉取指定好友](#)

拉取指定好友

最近更新时间：2018-04-19 18:34:11

功能说明

- 1.支持拉取指定好友；
- 2.建议每次拉取的好友数不超过 100，避免因回包数据量太大导致回包失败。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/sns/friend_get_list?usersig=xxx&identifier=admin&sdkappid=888888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

基础形式

```
{  
  "From_Account": "id",  
  "To_Account":  
  [  
    ]  
}
```

```
"id1"
]
```

完整形式

```
{
  "From_Account": "id",
  "To_Account": [
    "id1"
  ],
  "TagList": [
    "Tag_Profile_IM_Nick",
    "Tag_SNS_IM_Group",
    "Tag_SNS_IM_Remark"
  ]
}
```

批量拉取指定好友

```
{
  "From_Account": "id",
  "To_Account": [
    "id1",
    "id2",
    "id3"
  ],
  "TagList": [
    "Tag_Profile_IM_Nick",
    "Tag_SNS_IM_Group",
    "Tag_SNS_IM_Remark"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	请求拉取该Identifier的好友。

字段	类型	属性	说明
To_Account	Array	必填	请求拉取的好友的Identifier列表。 注意：每次拉取的好友数不得超过100，避免因回包数据量太大以致回包失败。
TagList	Array	选填	指定要拉取的资料字段及好友字段： 1、标配资料字段可参见 标配资料字段 ； 2、自定义资料字段可参见 自定义资料字段 ； 3、标配好友字段可参见 标配好友字段 ； 4、自定义好友字段可参见 自定义好友字段 。

应答

应答包体示例

基础形式、完备形式

```

{
  "Infoltem":
  [
    {
      "To_Account":"id1",
      "SnsProfileItem":
      [
        {
          "Tag":"Tag_Profile_IM_Nick",
          "Value":"NickTest1"
        },
        {
          "Tag":"Tag_SNS_IM_Group",
          "Value":["Group1"]
        },
        {
          "Tag":"Tag_SNS_IM_Remark",
          "Value":"Remark1"
        }
      ],
      "ResultCode": 0,
      "ResultInfo": ""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",

```

```

"ErrorCode":0,
"ErrorInfo":"","
"ErrorDisplay":""
}
    
```

批量拉取指定好友

```

{
  "Infoltem":
  [
    {
      "To_Account":"id1",
      "SnsProfileItem":
      [
        {
          "Tag":"Tag_Profile_IM_Nick",
          "Value":"NickTest1"
        },
        {
          "Tag":"Tag_SNS_IM_Group",
          "Value":["Group1"]
        },
        {
          "Tag":"Tag_SNS_IM_Remark",
          "Value":"Remark1"
        }
      ],
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account":"id2",
      "SnsProfileItem":
      [
        {
          "Tag":"Tag_Profile_IM_Nick",
          "Value":"NickTest2"
        },
        {
          "Tag":"Tag_SNS_IM_Group",
          "Value":["Group1","Group2"]
        },
        {
          "Tag":"Tag_SNS_IM_Remark",
          "Value":"Remark2"
        }
      ]
    }
  ]
}
    
```

```

    }
  ],
  "ResultCode": 0,
  "ResultInfo": ""
},
{
  "To_Account": "id3",
  "ResultCode": 31211,
  "ResultInfo": "SNS_FRD_GET_LIST_FRD_NO_EXIST"
}
],
"Fail_Account":["id3"],
"Invalid_Account":[],
"ActionStatus":"OK",
"ErrorCode":0,
"ErrorInfo": "",
"ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
Infoltem	Array	好友对象数组。
To_Account	String	好友的 Identifier。
SnsProfileItem	Array	好友的详细信息数组，数组每一个元素都包括 Tag 和 Value。
Tag	String	好友的资料字段或好友字段的名称。
Value	String	好友的资料字段或好友字段的值。
ResultCode	Integer	单个 Identifier 的处理结果，0 表示业务成功，非 0 表示业务失败。
ResultInfo	String	单个 Identifier 的详细错误信息，成功时该字段为空。
Fail_Account	Array	返回处理失败的用户列表。
Invalid_Account	Array	返回请求包中的非法用户列表。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。

字段	类型	说明
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整行； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段或联系技术客服。
31201	按列表拉好友请求的参数错误，请确认请求包的合法性或联系技术客服。
31202	请求拉取的字段数非法，请求拉取的关系链字段数不得超过 30 个，资料字段数不得超过 5000 个。
31203	按列表拉取好友主键不存在，请再次尝试或联系技术客服。
31204	按列表拉取好友关系链标配信息错误，请再次尝试或联系技术客服。
31205	按列表拉取好友关系链标配信息组数据错误，请再次尝试或联系技术客服。
31206	按列表拉取好友关系链标配信息行数据错误，请再次尝试或联系技术客服。
31207	按列表拉取好友关系链自定义信息错误，请再次尝试或联系技术客服。
31208	按列表拉取好友关系链自定义信息行数据错误，请再次尝试或联系技术客服。
31209	按列表拉取好友取执行任务失败，请再次尝试或联系技术客服。
31210	按列表拉取好友获取字段对应的字符串错误，请再次尝试或联系技术客服。
31211	按列表拉取好友不存在，拉取指定好友，但是指定好友并不在好友表中，调用方可以捕捉该错误给用户一个合理的提示。

错误码	详细错误信息
31212	拉取好友新建任务失败，请再次尝试或联系技术客服。
31213	拉取好友资料标配信息主键不存在，请再次尝试或联系技术客服。
31214	拉取好友资料标配信息错误，请再次尝试或联系技术客服。
31215	拉取好友资料标配信息行数据错误，请再次尝试或联系技术客服。
31216	拉取好友资料自定义信息错误，请再次尝试或联系技术客服。
31217	拉取好友资料自定义行数据错误，请再次尝试或联系技术客服。

接口调试工具

Web调试工具

通过 [REST API调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
// 获取用户指定好友  
./TimRestApiGear.php sns friend_get_list($api, $data_list)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 拉取某一用户的指定好友  
abstract function sns_friend_get_list($account_id, $frd_id);  
  
// 拉取某一用户的指定好友 (高级接口)  
abstract function sns_friend_get_list2($account_id, $frd_list, $tag_list);
```

可能触发的回调

无。

参考

REST API : [添加好友](#)

REST API : [导入好友](#)

REST API : [更新好友](#)

REST API : [删除好友](#)

REST API : [删除所有好友](#)

REST API : [校验好友](#)

REST API : [拉取好友](#)

添加黑名单

最近更新时间：2018-08-30 11:06:39

功能说明

1.支持批量添加黑名单。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/black_list_add?usersig=xxx&identifier=admin&sdkappid=888888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3"]
}
```

请求包字段说明

字段	类型	属性	说明
----	----	----	----

From_Account	String	必填	请求为该 Identifier 添加黑名单。
To_Account	Array	必填	待添加为黑名单的用户 Identifier 列表，单次请求的 To_Account 数不得超过 1000。

应答

应答包体示例

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id2",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id3",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"",
  "ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
ResultItem	Array	批量添加黑名单的结果对象数组。
To_Account	String	请求添加为黑名单的用户 Identifier。

ResultCode	Integer	To_Account 的处理结果，0 表示删除成功，非 0 表示删除失败。
ResultInfo	String	To_Account 的处理详情。
Fail_Account	Array	返回处理失败的 To_Account 列表。
Invalid_Account	Array	返回请求包中的非法 To_Account 列表。
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见[REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
31301	请求添加为黑名单的 To_Account 个数非法，详情可参见 请求包字段说明 。
31302	校验 SDKAppId 失败，请再次尝试或联系技术客服。
31303	读好友数据失败，请再次尝试或联系技术客服。
31304	解除好友关系失败，请再次尝试或联系技术客服。
31305	读黑名单数据失败，请再次尝试或联系技术客服。
31306	写黑名单数据失败，请再次尝试或联系技术客服。

31307	已经存在黑名单关系，调用方可以捕捉该错误码给用户一个合理的提示。
31308	获取黑名单总数错误，请再次尝试或联系技术客服。
31309	黑名单人数已达上限，调用方可以捕捉该错误码给用户一个合理的提示。
31310	并发添加黑名单导致写冲突，建议使用批量添加接口。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

可能触发的回调

参见 [添加黑名单之后回调](#)。

参考

REST API : [删除黑名单](#)

REST API : [拉取黑名单](#)

REST API : [校验黑名单](#)

删除黑名单

最近更新时间：2018-04-19 18:27:52

功能说明

删除指定黑名单。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/black_list_delete?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API 调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3"]
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要删除该 Identifier 的黑名单。
To_Account	Array	必填	待删除的黑名单的 Identifier 列表，单次请求的 To_Account 数不得超过 1000。

应答

应答包体示例

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id2",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id3",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo":"",
  "ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
ResultItem	Array	批量删除黑名单的结果对象数组。

字段	类型	说明
To_Account	String	请求删除的黑名单的 Identifier。
ResultCode	Integer	To_Account 的处理结果，0 表示删除成功，非 0 表示删除失败。
ResultInfo	String	To_Account 的处理详情。
Fail_Account	Array	返回处理失败的 To_Account 列表。
Invalid_Account	Array	返回请求包中的非法 To_Account 列表。
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
31501	请求删除的 To_Account 个数非法，详情可参见 请求包字段说明 。
31502	校验黑名单关系失败，请再次尝试或联系技术客服。
31503	To_Account 与 From_Account 之间不存在黑名单关系，调用方可以捕捉该错误码给用户一个合理的提示。
31504	删除黑名单行数据失败，请再次尝试或联系技术客服

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

可能触发的回调

参见 [删除黑名单之后回调](#)。

参考

REST API : [添加黑名单](#)

REST API : [拉取黑名单](#)

REST API : [校验黑名单](#)

拉取黑名单

最近更新时间：2018-05-23 16:40:00

功能说明

支持分页拉取所有黑名单。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/black_list_get?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "StartIndex": 0,
  "MaxLimited": 30,
  "LastSequence": 1431000000
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要拉取该 Identifier 的黑名单。
StartIndex	Integer	必填	拉取的起始位置。
MaxLimited	Integer	必填	每页最多拉取的黑名单数。
LastSequence	Integer	必填	上一次拉黑名单时后台返回给客户端的 Seq，初次拉取时为 0。

应答

应答包体示例

```

{
  "BlackListItem":
  [
    {
      "Black_Account":"id1",
      "BlackTimeStamp":1430000001
    },
    {
      "Black_Account":"id2",
      "BlackTimeStamp":1430000002
    }
  ],
  "StartIndex":0,
  "CurruentSequence":1431000001,
  "ActionStatus":"OK",
  "ErrorCode":0,
  "ErrorInfo": "",
  "ErrorDisplay":""
}
    
```

应答包字段说明

字段	类型	说明
BlackListItem	Array	黑名单对象数组，每一个黑名单对象都包括了 Black_Account 和 BlackTimeStamp。
Black_Account	String	黑名单的 Identifier。
BlackTimeStamp	Integer	添加黑名单的时间。

字段	类型	说明
StartIndex	Integer	下页拉取的起始位置，0 表示已拉完。
CurrentSequence	Integer	黑名单最新的 Seq。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
31601	请求拉取的黑名单数为 0 或超过 1000，请确认请求包中 MaxLimited 字段的合法性，如果没有问题，请联系技术客服。
31602	拉取黑名单读黑名单数据错误，请再次尝试或联系技术客服。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

可能触发的回调

无。

参考

REST API : [添加黑名单](#)

REST API : [删除黑名单](#)

REST API : [校验黑名单](#)

校验黑名单

最近更新时间：2018-04-19 18:28:24

功能说明

支持批量校验黑名单。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/black_list_check?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API调用频率调整](#) 提交工单申请处理。

HTTP请求方式

POST

请求

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "To_Account": ["id1", "id2", "id3", "id4"],
  "CheckType": "BlackCheckResult_Type_Both"
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要校验该 Identifier 的黑名单。
To_Account	Array	必填	待校验的黑名单的 Identifier 列表，单次请求的 To_Account 数不得超过 1000。
CheckType	String	必填	校验模式，详情可参见 校验黑名单 。

应答

应答包体示例

```

{
  "ResultItem":
  [
    {
      "To_Account":"id1",
      "Relation":"BlackCheckResult_Type_NO",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id2",
      "Relation":"BlackCheckResult_Type_AWithB",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id3",
      "Relation":"BlackCheckResult_Type_BWithA",
      "ResultCode":0,
      "ResultInfo":""
    },
    {
      "To_Account":"id4",
      "Relation":"BlackCheckResult_Type_BothWay",
      "ResultCode":0,
      "ResultInfo":""
    }
  ],
  "Fail_Account":[],
  "Invalid_Account":[],
  "ActionStatus":"OK",
  "ErrorCode":0,

```

```
"ErrorInfo": "",
"ErrorDisplay": ""
}
```

应答包字段说明

字段	类型	说明
CheckItem	Array	校验结果对象数组。
To_Account	String	请求校验的用户的 Identifier。
Relation	String	校验成功时 To_Account 与 From_Account 之间的黑名单关系，详情可参见 校验黑名单 。
ResultCode	Integer	To_Account 的处理结果，0 表示校验成功，非 0 表示校验失败。
ResultInfo	String	To_Account 的处理详情。
Fail_Account	Array	返回处理失败的 To_Account 列表。
Invalid_Account	Array	返回请求包中的非法 To_Account 列表。
ActionStatus	String	请求包的处理结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见[REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。

错误码	详细错误信息
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
31401	请求校验的 To_Account 个数非法，详情可参见 请求包字段说明 。
31403	校验正向黑名单失败，请再次尝试或联系技术客服。
31404	黑名单校验模式非法，详情可参见 请求包字段说明 。
31405	校验反向黑名单的任务失败，请再次尝试或联系技术客服。
31406	校验反向黑名单出错，请再次尝试或联系技术客服。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

可能触发的回调

无。

参考

REST API：[添加黑名单](#)

REST API：[删除黑名单](#)

REST API：[拉取黑名单](#)

添加分组

最近更新时间：2018-04-19 18:28:41

功能说明

批量添加分组，并将指定好友加入到新增分组中。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/sns/group_add?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[REST API调用频率调整](#)提交工单申请处理。

请求

HTTP请求方式

POST

HTTP请求包体格式

JSON

请求包示例

基础形式

```
{
  "From_Account": "id",
  "GroupName": ["group1", "group2", "group3"]
}
```

完整形式

```
{
  "From_Account": "id",
  "GroupName": ["group1", "group2", "group3"],
  "To_Account": ["id1", "id2", "id3", "id4"]
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要为该 Identifier 添加新分组。
GroupName	Array	必填	新增分组列表。
To_Account	Array	选填	需要加入新增分组的好友的 Identifier 列表。

应答

应答包体示例

基础形式

```
{
  "CurrentSequence": 2,
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "",
  "ErrorDisplay": ""
}
```

完备形式

```
{
  "ResultItem":
  [
    {
      "To_Account": "id1",
      "ResultCode": 32216,
      "ResultInfo": "Err_SNS_GroupAdd_ToTinyId_Not_Friend"
    },
    {
      "To_Account": "id2",
      "ResultCode": 32216,
      "ResultInfo": "Err_SNS_GroupAdd_ToTinyId_Not_Friend"
    }
  ]
}
```

```

    },
    {
      "To_Account": "id3",
      "ResultCode": 30002,
      "ResultInfo": "ERR_SDKAPPID_ILLEGAL"
    },
    {
      "To_Account": "id4",
      "ResultCode": 0,
      "ResultInfo": "OK"
    }
  ],
  "Fail_Account": ["id1", "id2"],
  "Fail_Account": ["id3"],
  "CurrentSequence": 3,
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "",
  "ErrorDisplay": ""
}
    
```

应答包字段说明

字段	类型	说明
ResultItem	Array	好友加入新增分组的结果对象数组。
To_Account	String	请求加入新分组的好友的 Identifier。
ResultCode	Integer	好友加入新分组的处理结果，0 表示业务成功，非 0 表示业务失败。
Fail_Account	Array	返回处理失败的用列表。
Invalid_Account	Array	返回请求包中的非法用户列表。
CurrentSequence	Integer	返回最新的分组 Sequence。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段或联系技术客服。
32201	请求添加的分组个数非法，分组个数的取值范围是：[0, 32]。
32202	请求添加的分组名长度非法，详情可参见 标配好友字段 。
32203	请求添加到分组的帐号个数非法，分组下帐号个数的取值范围是：[0, 1000]。
32204	加分组请求校验 SDKAppId 时 New Task 失败，请再次尝试或联系技术客服。
32205	加分组请求校验 SDKAppId 的 Task 出错，请再次尝试或联系技术客服。
32206	加分组请求校验 SDKAppId 失败，请再次尝试或联系技术客服。
32207	加分组请求没有拉到帐号对应的 SDKAppId，请再次尝试或联系技术客服。
32208	读分组数据失败，请再次尝试或联系技术客服。
32209	解分组行数据失败，请再次尝试或联系技术客服。
32210	解分组元数据失败，请再次尝试或联系技术客服。
32211	封装分组行数据失败，请再次尝试或联系技术客服。
32212	封装分组元数据失败，请再次尝试或联系技术客服。
32213	写分组数据失败，请再次尝试或联系技术客服。
32214	分组ID非法，详情可参见 标配好友字段 。
32215	分组个数已达系统上限，详情可参见 标配好友字段 。

错误码	详细错误信息
32216	与请求加入分组的帐号之间不存在好友关系，请确认帐号的合法性。
32217	请求加入分组的好友所在分组数超门限，请确认该好友所属的分组数，每个好友可归属的分组数的取值范围是：[0，32]。
32218	请求创建的分组已存在，请确认新分组的合法性。
32219	请求加入该分组的好友已存在，请确认好友帐号的合法性。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

API 集成

无。

可能触发的回调

无。

参考

REST API : [删除分组](#)

删除分组

最近更新时间：2018-04-19 18:34:46

功能说明

删除指定分组。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/sns/group_delete?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API 简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [REST API 调用频率调整](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "From_Account": "id",
  "GroupName": ["group1", "group2", "group3"]
}
```

请求包字段说明

字段	类型	属性	说明
From_Account	String	必填	需要删除该 Identifier 的分组。
GroupName	Array	必填	要删除的分组列表。

应答

应答包体示例

```
{
  "CurrentSequence": 4,
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "0",
  "ErrorDisplay": ""
}
```

应答包字段说明

字段	类型	说明
CurrentSequence	Integer	返回最新的分组Sequence。
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	详细错误信息。
ErrorDisplay	String	详细的客户端展示信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	详细错误信息
30001	关系链系统解包失败： 如果是 REST API 请求，请参照 REST API 文档确认请求包的合法性和完整性； 如果不是 REST API 请求，请联系技术客服。

错误码	详细错误信息
30002	SDKAppId 非法，请检查 SDKAppId 及帐号的合法性或联系技术客服。
30003	帐号非法，请检查帐号的合法性或联系技术客服。
30004	来自 REST API 的请求，但是不具备管理员权限，请联系技术客服。
30005	关系链字段中有敏感词，请检查关系链字段或联系技术客服。
32301	请求删除的分组个数非法，分组个数的取值范围是：[1, 32]。
32302	请求删除的分组名长度非法，分组名长度的取值范围是：[1, 30]。
32303	读分组数据失败，请再次尝试或联系技术客服。
32304	解分组行数据失败，请再次尝试或联系技术客服。
32305	解分组元数据失败，请再次尝试或联系技术客服。
32306	封装分组行数据失败，请再次尝试或联系技术客服。
32307	封装分组元数据失败，请再次尝试或联系技术客服。
32308	写分组数据失败，请再次尝试或联系技术客服。
32309	请求删除的分组不存在，请确认分组的合法性。

接口调试工具

通过 [REST API调试工具](#) 调试本接口。

API集成

无

可能触发的回调

无

参考

REST API : [添加分组](#)

群组管理

获取APP中的所有群组

最近更新时间：2018-05-25 18:19:19

功能说明

App 管理员可以通过该接口获取 App 中所有群组的 ID。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_appid_group_list?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 基础形式

如果 App 中的总群数量超过 10000 个，最多只会返回 10000 个群组 ID（如果需要完整获取，必须使用分页拉取的形式）。

```
{}
```

• 分页拉取

可以使用 Limit 和 Next 两个值用于控制分页拉取：

- i. Limit 限制回包中 GroupIdList 中群组的个数，不得超过 10000；
- ii. Next 控制分页。对于分页请求，第一次填 0，后面的请求填上一次返回的 Next 字段，当返回的 Next 为 0，代表所有的群都拉取到了；
- iii. 例如：假设需要分页拉取，每页展示 20 个，则第一页的请求参数应当为 { "Limit" : 20, "Next" : 0 }，第二页的请求参数应当为 { "Limit" : 20, "Next" : 上次返回的Next字段 }，依此类推；
- iv. Limit 或者 Next 的取值不会对应答包体中的 TotalCount 造成影响。

```
{  
  "Limit": 1000,  
  "Next": 0  
}
```

• 指定群组形态

可以指定拉取的群组所属的群组形态，如 Public，Private，ChatRoom、AVChatRoom和BChatRoom。

```
{  
  "GroupType": "Public" // 拉取哪种群组形态，不填为拉取所有  
}
```

• ALL IN ONE

```
{  
  "Limit": 1000,  
  "Next": 0,  
}
```

```
"GroupType": "Public" // 拉取哪种群组形态，不填为拉取所有
}
```

请求包字段说明

字段	类型	属性	说明
Limit	Integer	选填	本次获取的群组 ID 数量的上限，不得超过 10000。如果不填，默认为最大值 10000。
Next	Integer	选填	群太多时分页拉取标志，第一次填 0，以后填上一次返回的值，返回的 Next 为 0 代表拉完了。
GroupType	String	选填	如果仅需要返回特定群组形态的群组，可以通过 GroupType 进行过滤，但此时返回的 TotalCount 的含义就变成了 App 中属于该群组形态的群组总数。不填为获取所有类型的群组。 群组形态 包括 Public（公开群），Private（私密群），ChatRoom（聊天室），AVChatRoom（互动直播聊天室）和 BChatRoom（在线成员广播大群）。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "TotalCount": 2,
  "GroupIdList": [
    {
      "GroupId": "@TGS#2J4SZEAE"
    },
    {
      "GroupId": "@TGS#2C5SZEAEF"
    }
  ],
  "Next": 4454685361
}
```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
TotalCount	Integer	App 当前的群组总数。如果仅需要返回特定群组形态的群组，可以通过 GroupType 进行过滤，但此时返回的 TotalCount 的含义就变成了 App 中该群组形态的群组总数。 例如：假设 App 旗下总共 50000 个群组，其中有 20000 个为公开群组，如果将请求包体中的 GroupType 设置为 Public，那么不论 Limit 和 Offset 怎样设置，应答包体中的 TotalCount 都为 20000，且 GroupIdList 中的群组全部为公开群组。
GroupIdList	Array	获取到的群组 ID 的集合。
Next	Integer	分页拉取的标志。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10018	应答包长度超限。因为请求的内容过多，导致应答包超过了最大包长（1MB），请尝试减少单次请求的数据量。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取APP中所有群组信息(默认获取50个)
./TimRestApiGear.php group_open_http_svc get_appid_group_list
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 获取app中所有群组, 最多只会返回10000个(如果需要获取完整必须使用高级接口)
abstract function group_get_appid_group_list();

// 获取app中所有群组(高级接口)
abstract function group_get_appid_group_list2($limit, $offset, $group_type);
```

可能触发的回调

无。

参考

无。

创建群组

最近更新时间：2018-07-25 17:44:51

功能说明

App 管理员可以通过该接口创建群组。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：创建互动直播聊天室和在线成员广播大群时不能拉人入群，如果创建时指定了拉入的群成员会返回 10007 错误。用户加入互动直播聊天室和在线成员广播大群的唯一方式是用户申请入群。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/create_group?usersig=xxx&identifier=admin&sd  
kappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **基础形式**

创建群，Owner_Account 字段是选填的，如果不填则群没有群主。

```
{
  "Owner_Account": "leckie", // 群主的UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom/AVChatRoom/BChatRoom (必填)
  "Name": "TestGroup" // 群名称 (必填)
}
```

- **仅包含群基础信息**

创建群，并指定群简介、群公告等群基础信息。

```
{
  "Owner_Account": "leckie", // 群主的UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom/AVChatRoom/BChatRoom (必填)
  "Name": "TestGroup", // 群名称 (必填)
  "Introduction": "This is group Introduction", // 群简介 (选填)
  "Notification": "This is group Notification", // 群公告 (选填)
  "FaceUrl": "http://this.is.face.url", // 群头像URL (选填)
  "MaxMemberCount": 500, // 最大群成员数量 (选填)
  "ApplyJoinOption": "FreeAccess" // 申请加群处理方式 (选填)
}
```

- **仅包含群成员信息**

创建群，并指定初始化群成员列表，群成员列表在请求包说明表中有描述。

```
{
  "Name": "TestGroup", // 群名称 (必填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom(不支持AVChatRoom和BChatRoom) (必填)
}
```

```
"MemberList": [ // 初始群成员列表，最多500个（选填）
{
  "Member_Account": "bob", // 成员（必填）
  "Role": "Admin" // 赋予该成员的身份，目前备选项只有Admin（选填）
},
{
  "Member_Account": "peter"
}
]
```

• 自定义群组 ID

为了使得群组 ID 更加简单，腾讯云支持 App 在通过 REST API 创建群组时自定义群组 ID。

```
{
  "Owner_Account": "leckie", // 群主的UserId（选填）
  "Type": "Public", // 群组类型：Private/Public/ChatRoom/AVChatRoom/BChatRoom（必填）
  "GroupId": "MyFirstGroup", // 用户自定义群组ID（选填）
  "Name": "TestGroup" // 群名称（必填）
}
```

• 仅包含群维度自定义信息

创建群，并指定群维度的自定义字段。AppDefineData 默认是没有的，需要开通，详见请求包字段说明表。

```
{
  "Name": "TestGroup", // 群名称（必填）
  "Type": "Public", // 群组类型：Private/Public/ChatRoom/AVChatRoom/BChatRoom（必填）
  "AppDefinedData": [ // 群组维度的自定义字段（选填）
    {
      "Key": "GroupTestData1", // APP自定义的字段Key
      "Value": "xxxx" // 自定义字段的值
    },
    {
      "Key": " GroupTestData2",
      "Value": "abc\u0000\u0001" // 自定义字段支持二进制数据
    }
  ]
}
```

• 仅包含群成员维度自定义信息

创建群，并指定群成员维度的自定义字段。AppMemberDefinedData 默认是没有的，需要开通后才能使用，详

见请求包字段说明表。

```
{
  "Owner_Account": "leckie", // 群主的UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom(不支持AVChatRoom和BChatRoom) (必填)
  "Name": "TestGroup", // 群名称 (必填)
  "MemberList": [
    {
      "Member_Account": "bob",
      "AppMemberDefinedData": [ // 群成员维度自定义字段 (选填)
        {
          "Key": "MemberDefined1", // 群成员维度自定义的Key
          "Value": "MemberData1" // 群成员维度自定义字段值
        },
        {
          "Key": "MemberDefined2",
          "Value": "MemberData2"
        }
      ]
    },
    {
      "Member_Account": "peter",
      "AppMemberDefinedData": [
        {
          "Key": "MemberDefined1",
          "Value": "MemberData1"
        },
        {
          "Key": "MemberDefined2",
          "Value": "MemberData2"
        }
      ]
    }
  ]
}
```

• ALL IN ONE

```
{
  "Owner_Account": "leckie", // 群主的UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom(不支持AVChatRoom和BChatRoom) (必填)
  "GroupId": "MyFirstGroup", // 用户自定义群组ID (选填)
  "Name": "TestGroup", // 群名称 (必填)
  "Introduction": "This is group Introduction", // 群简介 (选填)
}
```

```
"Notification": "This is group Notification", // 群公告 (选填)
"FaceUrl": "http://this.is.face.url", // 群头像URL (选填)
"MaxMemberCount": 500, // 最大群成员数量 (选填)
"ApplyJoinOption": "FreeAccess", // 申请加群处理方式 (选填)
"AppDefinedData": [ // 群组维度的自定义字段 (选填)
{
"Key": "GroupTestData1", // APP自定义的字段Key
"Value": "xxxxx" // 自定义字段的值
},
{
"Key": "GroupTestData2",
"Value": "abc\u0000\u0001" // 自定义字段支持二进制数据
}
],
"MemberList": [ // 初始群成员列表, 最多500个 (选填)
{
"Member_Account": "bob", // 成员 (必填)
"Role": "Admin", // 赋予该成员的身份, 目前备选项只有Admin (选填)
"AppMemberDefinedData": [ // 群成员维度自定义字段 (选填)
{
"Key": "MemberDefined1", // 群成员维度自定义的Key
"Value": "MemberData1" // 群成员维度自定义字段值
},
{
"Key": "MemberDefined2",
"Value": "MemberData2"
}
]
},
{
"Member_Account": "peter",
"AppMemberDefinedData": [
{
"Key": "MemberDefined1",
"Value": "MemberData1"
},
{
"Key": "MemberDefined2",
"Value": "MemberData2"
}
]
}
]
}
```

请求包字段说明

字段	类型	属性	说明
Owner_Account	String	选填	群主 ID，自动添加到群成员中。如果不填，群没有群主。
Type	String	必填	群组形态 ，包括 Public（公开群），Private（私密群），ChatRoom（聊天室），AVChatRoom（互动直播聊天室），BChatRoom（在线成员广播大群）。
GroupId	String	选填	为了使得群组 ID 更加简单，便于记忆传播，腾讯云支持 App 在通过 REST API 创建群组时自定义群组 ID。详情参见： 自定义群组ID 。
Name	String	必填	群名称，最长 30 字节。
Introduction	String	选填	群简介，最长 240 字节。
Notification	String	选填	群公告，最长 300 字节。
FaceUrl	String	选填	群头像 URL，最长 100 字节。
MaxMemberCount	Integer	选填	最大群成员数量，缺省时的默认值：私有群是 200，公开群是 2000，聊天室是 10000，互动直播聊天室和在线成员广播大群无限制。
ApplyJoinOption	String	选填	申请加群处理方式。包含 FreeAccess（自由加入），NeedPermission（需要验证），DisableApply（禁止加群），不填默认为 NeedPermission（需要验证）。
AppDefinedData	Array	选填	群组维度的自定义字段，默认情况是没有的，需要开通，详情参见： 自定义字段 。
MemberList	Array	选填	初始群成员列表，最多 500 个。成员信息字段详情参见： 群成员资料 。
AppMemberDefinedData	Array	选填	群成员维度的自定义字段，默认情况是没有的，需要开通，详情参见： 自定义字段 。

应答

应答包体示例

- 基础形式、仅包含群基础信息、仅包含群成员信息和仅包含自定义信息

```
{
  "ActionStatus": "OK",
  "ErrorInfo": ""
}
```

```
"ErrorCode": 0,
"GroupId": "@TGS#2J4SZEAEEL"
}
```

• 自定义群组ID和ALL IN ONE

```
{
"ActionStatus": "OK",
"ErrorInfo": "",
"ErrorCode": 0,
"GroupId": "MyFirstGroup"
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
GroupId	String	创建成功之后的群 ID，由 IM 云后台分配。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。

错误码	含义说明
10004	<p>参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。优先自查：</p> <ol style="list-style-type: none"> 1. Owner_Account 指定的群组是否已导入； 2. 没有填 Name 指定的群名、或群名为空、或者长度超过 30 字节； 3. 没有填 Type 指定的群类型、或者类型不是标准的 Public Private ChatRoom AVChatRoom BChatRoom、或者类型也不是通过 工单 创建的自定义群组类型； 4. Introduction 指定的群简介长度不能超过 240 字节； 5. Notification 指定的群公告长度不能超过 300 字节； 6. FaceUrl 指定的群头像 URL 长度不能超过 100 字节； 7. GroupId 指定的自定义群 ID 长度不能超过 49 字节，不能包含非打印字符，前缀不能是"@TGS#"； 8. MaxMemberCount指定的群人数上限不能超过群形态支持的最大人数。Private最大支持200人，Public最大支持2000人，ChatRoom最大支持10000人，AVChatRoom和BChatRoom无上限限制； 9. ApplyJoinOption 指定的加群选项必须是 "DisableApply""NeedPermission""FreeAccess" 三者之一； 10. 群维度的自定义字段无效，包括 Key 不能为空且长度不能超过 16 字节，Key 不能重复，Value 长度不能超过 512 字节，其它情况请再次尝试或联系技术客服； 11. 参数中如果携带了 MemberList，则它必须是数组形式； 12. 邀请用户的个数不能超过群人数上限； 13. 群成员维度的自定义字段无效，包括 Key 不能为空且长度不能超过 16 字节，Key 不能重复，Value 长度不能超过 64 字节，其它情况请再次尝试或联系技术客服； 14. 目前被邀请用户中的 Role 字段只支持 "Admin"，非管理员不要保留该字段； 15. 群形态不支持添加管理员时，邀请用户不要保留 Role 字段； 16. 群主不能在邀请列表中赋予管理员的角色，即不要保留 Role 字段；
10005	请求包体中携带的用户数量过多（超过了 500 个成员），请减少一次请求的用户数量。
10006	<p>创建群组的个数超过配额，分为两种情况。</p> <ol style="list-style-type: none"> 1. 非 BChatRoom 每天创建群的个数有上限； 2. 同一个 sdkappid 下，创建 BChatRoom 的个数有上限；
10007	<p>操作权限不足。优先自查：</p> <ol style="list-style-type: none"> 1. 操作者是否是 App 管理员； 2. 群形态配置是否建群时也不允许直接拉人，如不确定请联系技术客服； 3. 群维度自定义字段或群成员自定义字段是否不允许任何人写，如不确定请联系技术客服；
10008	非法请求。优先自查建群者是否不能修改自己的群成员字段，如不确定请联系技术客服；
10016	该请求触发了到 App 后台的“ 创建群组之前回调 ”，云通讯后台根据回调结果拒绝创建该群组。
10021	群组 ID 已被使用，请选择其他的群组 ID。
80001	文本安全打击。请检查群名称、群公告和群简介等是否有敏感词汇。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 创建群组(最大成员数量默认为 500)
./TimRestApiGear.php group_open_http_svc create_group (group_type) (group_name) (owner_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 创建群
abstract function group_create_group($group_type, $group_name, $owner_id);

// 创建群(高级接口)
abstract function group_create_group2($group_type, $group_name, $owner_id,
$info_set, $mem_list);
```

可能触发的回调

[创建群组之前回调](#)；

[创建群组之后回调](#)。

参考

REST API：[解散群组 \(v4/group_open_http_svc/destroy_group\)](#)。

获取群组详细资料

最近更新时间：2018-05-25 18:19:58

功能说明

App 管理员可以根据群组ID获取群组的详细信息。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_group_info?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 基础形式

用来获取群组的详细信息（所有群基础信息、群成员信息和自定义信息），请求中只包含群 ID 的列表，一次请求最多支持 50 个群组。

```
{
  "GroupIdList": [ // 群组列表（必填）
    "@TGS#1NVTZAE4",
    "@TGS#1CXTZAE4"
  ]
}
```

• 拉取指定的信息

可以指定拉取的基础信息字段，在 GroupBaseInfoFilter 中设置；

可以指定拉取的成员信息字段，在 MemberInfoFilter 中设置；

可以指定拉取的群组维度的自定义字段，在 AppDefinedDataFilter_Group 中设置（该字段默认没有，需要额外开通，请求包字段表中有说明）；

可以指定拉取的群成员维度的自定义字段，在 AppDefinedDataFilter_GroupMember（该字段默认没有，需要额外开通，请求包字段表中有说明）；

如果设置了过滤器，没有在过滤器中指定的字段将不被拉取。

```
{
  "GroupIdList": [ // 群组列表，必填
    "@TGS#1NVTZAE4",
    "@TGS#1CXTZAE4"
  ],
  "ResponseFilter": { // 返回
    "GroupBaseInfoFilter": [ // 如果基础信息字段，请在此数组中添加
      "Type",
      "Name",
      "Introduction",
      "Notification"
    ],
    "MemberInfoFilter": [ // 如果需要成员信息，请添加此数组
      "Account", // 成员ID
      "Role"
    ],
    "AppDefinedDataFilter_Group": [ // 群组维度的自定义字段过滤，该默认情况是没有的

```

```

"GroupTestData1",
"GroupTestData2"
],
"AppDefinedDataFilter_GroupMember": [ //群成员维度自定义字段过滤, 该默认情况是没有的
"MemberDefined2",
"MemberDefined1"
]
}
}

```

请求包字段说明

字段	类型	属性	说明
GroupIdList	String	必填	需要拉取的群组列表。
ResponseFilter	Object	选填	包含三个过滤器：GroupBaseInfoFilter，MemberInfoFilter，AppDefinedDataFilter_Group，分别是基础信息字段过滤器，成员信息字段过滤器，群组维度的自定义字段过滤器。
GroupBaseInfoFilter	Array	选填	基础信息字段过滤器，指定需要获取的基础信息字段，基础信息字段详情参见： 群基础资料 。
MemberInfoFilter	Array	选填	成员信息字段过滤器，指定需要获取的成员信息字段，成员信息字段详情参见： 群成员资料 。
AppDefinedDataFilter_Group	Array	选填	该字段用来群组维度的自定义字段过滤器，指定需要获取的 群组维度的自定义字段 。
AppDefinedDataFilter_GroupMember	Array	选填	该字段用来群成员维度的自定义字段过滤器，指定需要获取的 群成员维度的自定义字段 。

应答

应答包体示例

- 基础形式

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "", //这里的 ErrorInfo 无意义, 需要判断每个群组的 ErrorInfo
  "ErrorCode": 0, // 这里的 ErrorCode 无意义, 需要判断每个群组的 ErrorCode
}

```

```
"GroupInfo": [ // 返回结果为群组信息数组，为简单起见这里仅列出一个群
{
  "GroupId": "@TGS#2J4SZEAEEL",
  "ErrorCode": 0, // 针对该群组的返回结果
  "ErrorInfo": "", // 针对该群组的返回结果
  "Type": "Public", // 群组类型
  "Name": "MyFirstGroup", // 群组名称
  "Introduction": "TestGroup", // 群组简介
  "Notification": "TestGroup", // 群组通知
  "FaceUrl": "http://this.is.face.url", // 群组头像
  "Owner_Account": "leckie", // 群主 ID
  "CreateTime": 1426976500, // 群组创建时间 ( UTC 时间 )
  "LastInfoTime": 1426976500, // 最后群资料变更时间 ( UTC 时间 )
  "LastMsgTime": 1426976600, // 群内最后一条消息的时间 ( UTC 时间 )
  "NextMsgSeq": 1234, //
  "MemberNum": 2, // 当前群成员数量
  "MaxMemberNum": 50, // 最大群成员数量
  "ApplyJoinOption": "FreeAccess", // 申请加群处理方式
  "ShutUpAllMember": "On", // 群全员禁言状态
  "AppDefinedData": [ // 群组维度的自定义字段
  {
    "Key": "GroupTestData1", // 自定义字段的key
    "Value": "xxxx" // 自定义字段的值
  },
  {
    "Key": "GroupTestData2",
    "Value": "abc\u0000\u0001" // 自定义字段支持二进制数据
  }
  ],
  "MemberList": [ // 群成员列表
  {
    "Member_Account": "leckie", // 成员ID
    "Role": "Owner", // 群内角色
    "JoinTime": 1425976500, // 入群时间 ( UTC时间 )
    "MsgSeq": 1233,
    "MsgFlag": "AcceptAndNotify", // 消息屏蔽选项
    "LastSendMsgTime": 1425976500, // 最后发言时间 ( UTC时间 )
    "ShutUpUntil": 1431069882, // 禁言截至时间 ( UTC时间 )
    "AppMemberDefinedData": [ // 群成员自定义字段
    {
      "Key": "MemberDefined1",
      "Value": "ModifyDefined1"
    },
    {
      "Key": "MemberDefined2",
```

```
"Value": "ModifyDefined2"
}
],
{
  "Member_Account": "peter",
  "Role": "Member",
  "JoinTime": 1425976500, // 入群时间
  "MsgSeq": 1233,
  "MsgFlag": "AcceptAndNotify",
  "LastSendMsgTime": 1425976500, // 最后一次发消息的时间
  "ShutUpUntil": 0, // 0表示未被禁言, 否则为禁言的截止时间
  "AppMemberDefinedData": [ // 群成员自定义字段
    {
      "Key": "MemberDefined1",
      "Value": "ModifyDefined1"
    },
    {
      "Key": "MemberDefined2",
      "Value": "ModifyDefined2"
    }
  ]
}
]
```

- 拉取指定的信息

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "", // 这里的 ErrorInfo 无意义, 需要判断每个群组的 ErrorInfo
  "ErrorCode": 0, // 这里的 ErrorCode 无意义, 需要判断每个群组的 ErrorCode
  "GroupInfo": [ // 返回结果为群组信息数组, 为简单起见这里仅列出一个群
    {
      "GroupId": "@TGS#2J4SZEAE",
      "ErrorCode": 0, // 针对该群组的返回结果
      "ErrorInfo": "", // 针对该群组的返回结果
      "Type": "Public", // 群组类型
      "Name": "MyFirstGroup", // 群组名称
      "Introduction": "TestGroup", // 群组简介
      "Notification": "TestGroup", // 群组通知
      "AppDefinedData": [ // 群组维度的自定义字段
        {
          "Key": "GroupTestData1", // 自定义字段的 key

```

```

"Value": "xxxx" // 自定义字段的值
},
{
"Key": "GroupTestData2",
"Value": "abc\u0000\u0001" // 自定义字段支持二进制数据
}
],
"MemberList": [ // 群成员列表
{
"Member_Account": "leckie", // 成员 ID
"Role": "Owner", // 群内角色
"AppMemberDefinedData": [ // 群成员自定义字段
{
"Key": "MemberDefined1",
"Value": "ModifyDefined1"
},
{
"Key": "MemberDefined2",
"Value": "ModifyDefined2"
}
]
},
{
"Member_Account": "peter",
"Role": "Member",
"AppMemberDefinedData": [ // 群成员自定义字段
{
"Key": "MemberDefined1",
"Value": "ModifyDefined1"
},
{
"Key": "MemberDefined2",
"Value": "ModifyDefined2"
}
]
}
]
}
]
}
]
}
}

```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
GroupInfo	Array	返回结果为群组信息数组，内容包括 群基础资料 字段、 群成员资料 字段、 群组维度自定义字段 和 群成员维度自定义字段 。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10018	应答包长度超限。因为请求的内容过多，导致应答包超过了最大包长（1MB），请尝试减少单次请求的数据量。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取群组详细信息  
./TimRestApiGear.php group_open_http_svc get_group_info (group_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 获取群组详细信息  
abstract function group_get_group_info($group_id);  
  
// 获取群组详细信息(高级接口)  
abstract function group_get_group_info2($group_list, $base_info_filter, $member_info_filter,  
$app_define_filter);
```

可能触发的回调

无。

参考

REST API：获取群成员详细资料 ([v4/group_open_http_svc/get_group_member_info](#))；

REST API：修改群组基础资料 ([v4/group_open_http_svc/modify_group_base_info](#))；

REST API：修改群成员资料 ([v4/group_open_http_svc/modify_group_member_info](#))。

获取群组成员详细资料

最近更新时间：2018-05-25 18:44:59

功能说明

App 管理员可以根据群组 ID 获取群组成员的资料。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	300 人以内 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：因为内部实现的差异，互动直播聊天室只能获取 300 人以内的群成员资料，群人数达到 300 后加入的群成员的成员资料无法被获取；而在线成员广播大群则无法获取群成员资料。

请求URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_group_member_info?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 基础形式

用来获取群成员详细信息（群成员资料和群成员维度自定义字段），请求中只包含群 ID。

```
{
  "GroupId": "@TGS#1NVTZEAE4" // 群组ID (必填)
}
```

• 分页获取

可以使用 Limit 和 Offset 两个值用于控制分页拉取；

Limit 限制回包中 MemberList 数组中成员的个数，不得超过 10000；

Offset 控制从群成员中的第多少个成员开始拉取信息。对于分页请求（页码数字从1开始），每一页的 Offset 值应当为： $(\text{页码数} - 1) \times \text{每页展示的群成员数量}$ ；

例如：假设需要分页拉取，每页展示 20 个，则第一页的请求参数应当为：`{ "Limit" : 20, "Offset" : 0 }`，第二页的请求参数应当为：`{ "Limit" : 20, "Offset" : 20 }`，依此类推；

Limit 或者 Offset 的取值不会对应答包体中的 TotalCount 造成影响。

```
{
  "GroupId": "@TGS#1NVTZEAE4", // 群组ID (必填)
  "Limit": 100, // 最多获取多少个成员的资料
  "Offset": 0 // 从第多少个成员开始获取资料
}
```

• 指定拉取的信息

通过 MemberInfoFilter 过滤器字段选择需要拉取的字段。没有在过滤器中指定的字段将不被拉取。

```
{
  "GroupId": "@TGS#1NVTZEAE4", // 群组ID (必填)
  "MemberInfoFilter": [ // 需要获取哪些信息 (Member_Account被默认包含在其中)，如果没有该字段则为群成员全部资料
    "Role",
    "JoinTime",
    "MsgSeq",
  ]
}
```

```
"MsgFlag",
"LastSendMsgTime",
"ShutUpUntil",
"NameCard"
]
}
```

- **拉取指定身份成员**

通过 MemberRoleFilter 过滤器字段选择需要拉取资料的成员身份。没有在过滤器中指明则代表拉取任何身份的成员的资料。

```
{
  "GroupId": "@TGS#37AB3PAEC", // 群组ID (必填)
  "MemberRoleFilter": [ // 群成员身份过滤器
    "Owner",
    "Member"
  ]
}
```

- **拉取群成员自定义字段**

通过 AppDefinedDataFilter_GroupMember 过滤器选取需要拉取的成员自定义字段。没有在过滤器中指定的字段将不被拉取。

```
{
  "GroupId": "@TGS#37AB3PAEC", // 群组ID (必填)
  "AppDefinedDataFilter_GroupMember": [ // 群成员自定义字段过滤器
    "MemberDefined2" // 群成员自定义字段Key
  ]
}
```

- **ALL IN ONE**

```
{
  "GroupId": "@TGS#1NVTZAE4", // 群组ID (必填)
  "MemberInfoFilter": [ // 需要获取哪些信息, 如果没有该字段则为群成员全部资料
    "Role",
    "JoinTime",
    "MsgSeq",
    "MsgFlag",
    "LastSendMsgTime",
    "ShutUpUntil",
    "NameCard"
  ],
  "MemberRoleFilter": [ // 群成员身份过滤器
    "Owner",
```

```

"Member"
],
"AppDefinedDataFilter_GroupMember": [ //群成员自定义字段过滤器
"MemberDefined2", //群成员自定义字段Key
"MemberDefined1"
],
"Limit": 100, // 最多获取多少个成员的资料
"Offset": 0 // 从第多少个成员开始获取
}
    
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	需要拉取成员信息的群组的 ID。
MemberInfoFilter	Array	选填	需要获取哪些信息，如果没有该字段则为群成员全部资料，成员信息字段详情参见： 群成员资料 。
MemberRoleFilter	Array	选填	拉取指定身份的群成员资料。如没有填写该字段，默认为所有身份成员资料，成员身份可以为：“Owner”，“Admin”，“Member”。
AppDefinedDataFilter_GroupMember	Array	选填	默认情况是没有的。该字段用来群成员维度的自定义字段过滤器，指定需要获取的群成员维度的自定义字段，群成员维度的自定义字段详情参见： 自定义字段 。
Limit	Integer	选填	一次最多获取多少个成员的资料，不得超过 10000。如果不填，则获取群内全部成员的信息。
Offset	Integer	选填	从第几个成员开始获取，如果不填则默认为 0，表示从第一个成员开始获取。

应答

应答包体示例

- 基本形式和分页拉取

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
    
```

```
"MemberNum": 2, // 本群组的群成员总数
"MemberList": [ // 群成员列表
{
"Member_Account": "bob",
"Role": "Owner",
"JoinTime": 1425976500, // 入群时间
"MsgSeq": 1233,
"MsgFlag": "AcceptAndNotify",
"LastSendMsgTime": 1425976500, // 最后一次发消息的时间
"ShutUpUntil": 1431069882, // 禁言截至时间 (秒数)
"AppMemberDefinedData": [ //群成员自定义字段
{
"Key": "MemberDefined1",
"Value": "ModifyDefined1"
},
{
"Key": "MemberDefined2",
"Value": "ModifyDefined2"
}
]
},
{
"Member_Account": "peter",
"Role": "Member ",
"JoinTime": 1425976500,
"MsgSeq": 1233,
"MsgFlag": "AcceptAndNotify",
"LastSendMsgTime": 1425976500,
"ShutUpUntil": 0, // 0表示未被禁言, 否则为禁言的截止时间
"AppMemberDefinedData": [ //群成员自定义字段
{
"Key": "MemberDefined1",
"Value": "ModifyDefined1"
},
{
"Key": "MemberDefined2",
"Value": "ModifyDefined2"
}
]
}
]
}
```

- 拉取指定字段

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MemberNum": 2, // 本群组的群成员总数
  "MemberList": [ // 群成员列表
    {
      "Member_Account": "bob",
      "Role": "Owner",
      "JoinTime": 1425976500, // 入群时间
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500, // 最后一次发消息的时间
      "ShutUpUntil": 1431069882, // 禁言截至时间 (秒数)
    },
    {
      "Member_Account": "peter",
      "Role": "Member",
      "JoinTime": 1425976500,
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500,
      "ShutUpUntil": 0, // 0表示未被禁言, 否则为禁言的截止时间
    }
  ]
}
```

- 拉取指定身份成员

```
{
  "ActionStatus": "OK", //返回成功
  "ErrorCode": 0, //返回码
  "MemberList": [
    {
      "JoinTime": 1450680436, //成员加入时间
      "LastSendMsgTime": 0, //成员最后发消息时间
      "Member_Account": "Test_1", //成员帐号
      "MsgFlag": "AcceptNotNotify", //成员消息屏蔽类型
      "MsgSeq": 1, //成员已读消息seq
      "NameCard": "", //成员名片
      "Role": "Owner", // 成员身份
      "ShutUpUntil": 0 // 0表示未被禁言, 否则为禁言的截止时间
    },
    {
      "JoinTime": 1450680436,
```

```
"LastSendMsgTime": 0,
"Member_Account": "Test_6",
"MsgFlag": "AcceptNotNotify",
"MsgSeq": 1,
"NameCard": "",
"Role": "Admin",
"ShutUpUntil": 0
},
],
"MemberNum": 8 //本群组，群成员总数
}
```

- 拉取群成员自定义字段

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MemberNum": 2, // 本群组的群成员总数
  "MemberList": [ // 群成员列表
    {
      "Member_Account": "bob",
      "Role": "Owner",
      "JoinTime": 1425976500, // 入群时间
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500, // 最后一次发消息的时间
      "ShutUpUntil": 1431069882, // 禁言截至时间（秒数）
      "AppMemberDefinedData": [ // 群成员自定义字段
        {
          "Key": "MemberDefined2",
          "Value": "ModifyDefined2"
        }
      ]
    },
    {
      "Member_Account": "peter",
      "Role": "Member",
      "JoinTime": 1425976500,
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500,
      "ShutUpUntil": 0, // 0表示未被禁言，否则为禁言的截止时间
      "AppMemberDefinedData": [ // 群成员自定义字段
        {
          "Key": "MemberDefined2",
```

```

"Value": "ModifyDefined2"
}
]
}
]
}

```

• ALL IN ONE

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MemberNum": 2, // 本群组的群成员总数
  "MemberList": [ // 群成员列表
    {
      "Member_Account": "bob",
      "Role": "Owner",
      "JoinTime": 1425976500, // 入群时间
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500, // 最后一次发消息的时间
      "ShutUpUntil": 1431069882, // 禁言截至时间 (秒数)
      "AppMemberDefinedData": [ // 群成员自定义字段
        {
          "Key": "MemberDefined1",
          "Value": "ModifyDefined1"
        },
        {
          "Key": "MemberDefined2",
          "Value": "ModifyDefined2"
        }
      ]
    },
    {
      "Member_Account": "peter",
      "Role": "Member",
      "JoinTime": 1425976500,
      "MsgSeq": 1233,
      "MsgFlag": "AcceptAndNotify",
      "LastSendMsgTime": 1425976500,
      "ShutUpUntil": 0, // 0表示未被禁言, 否则为禁言的截止时间
      "AppMemberDefinedData": [ // 群成员自定义字段
        {
          "Key": "MemberDefined1",
          "Value": "ModifyDefined1"
        }
      ]
    }
  ]
}

```

```

    },
    {
      "Key": "MemberDefined2",
      "Value": "ModifyDefined2"
    }
  ]
}
]
}
}

```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
MemberNum	Integer	本群组的群成员总数。
MemberList	Array	获取到的群成员列表，其中包含了全部或者指定的群成员信息，成员信息字段详情参见： 群成员资料数据结构 。
AppMemberDefinedData	Array	返回的群成员自定义字段信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10005	请求包体中携带的用户数量过多（超过了 50 个成员），请减少一次请求的用户数量。
10007	操作权限不足。请确认操作者是否是 App 管理员。

错误码	含义说明
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10018	应答包长度超限。因为请求的内容过多，导致应答包超过了最大包长（1MB），请尝试减少单次请求的数据量。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取群组成员详细信息  
./TimRestApiGear.php group_open_http_svc get_group_member_info (group_id) (limit) (offset)
```

更多调试工具参见[REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 获取群组成员详细信息  
abstract function group_get_group_member_info($group_id, $limit, $offset);
```

可能触发的回调

无。

参考

REST API : 修改群成员资料 ([v4/group_open_http_svc/modify_group_member_info](#)) 。

修改群组基础资料

最近更新时间：2018-05-25 18:33:47

功能说明

App 管理员可以通过该接口修改指定群组的基础信息。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/modify_group_base_info?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例**• 修改群组基础信息**

修改群组基础信息，如群名称，群公告等。

```
{
  "GroupId": "@TGS#1NVTZAE4", // 要修改哪个群的基础资料 (必填)
  "Name": "NewName", // 群名称 (选填)
  "Introduction": "NewIntroduction", // 群简介 (选填)
  "Notification": "NewNotification", // 群公告 (选填)
  "FaceUrl": "http://this.is.new.face.url", // 群头像 (选填)
  "MaxMemberNum": 500, // 最大群成员数量 (选填)
  "ApplyJoinOption": "NeedPermission", // 申请加群方式 (选填)
  "ShutUpAllMember": "On" // 设置全员禁言 (选填): "On"开启, "Off"关闭
}
```

• 设置群自定义信息

设置群组自定义字段信息，自定义信息默认没有，需要开通，详见请求包字段说明表。

```
{
  "GroupId": "@TGS#1NVTZAE4", // 要修改哪个群的基础资料 (必填)
  "AppDefinedData": [ // 自定义字段 (选填)
    {
      "Key": "GroupTestData1", // 需要修改的自定义字段key
      "Value": "NewData" // 自定义字段的新值
    }
  ]
}
```

• 删除群自定义信息

删除已设置的群组自定义字段信息。

```
{
  "GroupId": "@TGS#1NVTZAE4", // 要修改哪个群的基础资料 (必填)
  "AppDefinedData": [ // 自定义字段 (选填)
    {
```

```

"Key": "GroupTestData2",
"Value": "" // 设置为空表示删除该项自定义字段
}
]
}
    
```

• ALL IN ONE

```

{
  "GroupId": "@TGS#2J4SZEAEEL", // 要修改哪个群的基础资料 (必填)
  "Name": "NewName", // 群名称 (选填)
  "Introduction": "NewIntroduction", // 群简介 (选填)
  "Notification": "NewNotification", // 群公告 (选填)
  "FaceUrl": "http://this.is.new.face.url", // 群头像 (选填)
  "MaxMemberNum": 500, // 最大群成员数量 (选填)
  "ApplyJoinOption": "NeedPermission", // 申请加群方式 (选填)
  "ShutUpAllMember": "On", // 设置全员禁言 (选填): "On"开启, "Off"关闭
  "AppDefinedData": [ // 自定义字段 (选填)
    {
      "Key": "GroupTestData1", // 需要修改的自定义字段key
      "Value": "NewData" // 自定义字段的新值
    },
    {
      "Key": "GroupTestData2",
      "Value": "" // 设置为空表示删除该项自定义字段
    }
  ]
}
    
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	需要修改基础信息的群组的 ID。
Name	String	选填	群名称，最长 30 字节。
Introduction	String	选填	群简介，最长 240 字节。
Notification	String	选填	群公告，最长 300 字节。
FaceUrl	String	选填	群头像 URL，最长 100 字节。
MaxMemberNum	Integer	选填	最大群成员数量，最大为 10000。

字段	类型	属性	说明
ApplyJoinOption	String	选填	申请加群处理方式。包含 FreeAccess (自由加入) , NeedPermission (需要验证) , DisableApply (禁止加群)。
AppDefinedData	Array	选填	默认情况是没有的。开通群组维度的自定义字段详情参见： 自定义字段 。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。

错误码	含义说明
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
80001	文本安全打击。请检查群名称、群公告和群简介等是否有敏感词汇。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的REST API调试，命令如下：

```
# 修改群组名称  
./TimRestApiGear.php group_open_http_svc modify_group_base_info (group_id) (group_name)
```

更多调试工具参见[REST API调试工具](#)。

API 集成

PHP集成

在云通信[PHP Server SDK](#)中，可以通过TimRestInterface的如下成员函数快速集成该API：

```
// 修改群组名称  
abstract function group_modify_group_base_info($group_id, $group_name);  
  
// 修改群组信息(高级接口)  
abstract function group_modify_group_base_info2($group_id, $group_name, $info_set,  
$app_define_list);
```

可能触发的回调

无。

参考

无。

增加群组成员

最近更新时间：2018-05-25 18:34:07

功能说明

App 管理员可以通过该接口向指定的群中添加新成员。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不支持增加群成员，对这两种类型的群组进行操作时会返回 10007 错误。用户加入这两种群组的唯一方式是用户申请加群。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/add_group_member?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 基础形式

用来向群中进行邀请加入，一次请求最多支持添加 500 个成员；后台默认情况下会给群中所有成员下发加群系统通知（除没有激活的 Private 类型群组外）。

```
{
  "GroupId": "@TGS#2J4SZEAE", // 要操作的群组 (必填)
  "MemberList": [ // 一次最多添加500个成员
    {
      "Member_Account": "tommy" // 要添加的群成员ID (必填)
    },
    {
      "Member_Account": "jared"
    }
  ]
}
```

• 静默加入

当 Silence 为 1 时，成员添加成功后，不会给任何人下发系统通知。

```
{
  "GroupId": "@TGS#2J4SZEAE", // 要操作的群组 (必填)
  "Silence": 1, // 是否静默加入 (选填)
  "MemberList": [ // 一次最多添加500个成员
    {
      "Member_Account": "tommy" // 要添加的群成员ID (必填)
    },
    {
      "Member_Account": "jared"
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。
Silence	Integer	选填	是否静默加入。0：非静默加入；1：静默加入。不填该字段默认为 0。
MemberList	Array	必填	待添加的群成员数组。
Member_Account	String	必填	待添加的群成员帐号。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MemberList": [
    {
      "Member_Account": "tommy",
      "Result": 1 // 加入结果：0 为失败；1 为成功；2 为已经是群成员
    },
    {
      "Member_Account": "jared",
      "Result": 1
    }
  ]
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
MemberList	Array	返回添加的群成员结果。
Member_Account	String	返回的群成员帐号。

字段	类型	说明
Result	Integer	加人结果：0-失败；1-成功；2-已经是群成员；3-等待被邀请者确认。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10005	请求包体中携带的用户数量过多（超过了 500 个成员），请减少一次请求的用户数量。
10007	操作权限不足。请确认操作者是否是APP管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10016	该请求触发了到 App 后台的“ 拉人入群之前回调 ”，云通讯后台根据回调结果拒绝增加该成员。
10019	被添加用户的帐号不存在，请检查用户帐号是否正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 添加群组成员
./TimRestApiGear.php group_open_http_svc add_group_member (group_id) (member_id) (silence)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 `TimRestInterface` 的如下成员函数快速集成该 API：

```
// 增加群组成员  
abstract function group_add_group_member($group_id, $member_id, $silence);
```

可能触发的回调

[拉人入群之前回调](#)；
[新成员入群之后回调](#)；
[群组满员之后回调](#)。

参考

REST API：删除群组成员（[v4/group_open_http_svc/delete_group_member](#)）。

删除群组成员

最近更新时间：2018-05-25 18:34:26

功能说明

App 管理员可以通过该接口删除群成员。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明 1)
在线成员广播大群 (BChatRoom)	否 (见说明 2)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：

- 互动直播聊天室不支持删除群成员，对这种类型的群组进行操作时会返回 10004 错误。如果管理员希望达到删除群成员的效果，可以通过 [设置禁言](#) 的方式实现。
- 在线成员广播大群也不支持删除群成员，对这种类型的群组进行操作时同样会返回 10004 错误。且该类型群组只支持 App 管理员推送广播消息。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/delete_group_member?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **基础形式**

用来向群中进行删除群成员，一次请求最多支持删除 500 个成员；后台默认情况下会给群中所有成员下发删除群成员系统通知（除没有激活的 Private 类型群组外）；

如果删除的用户原本就不在群组中，该接口依然返回成功。

```
{
  "GroupId": "@TGS#2J4SZEAE", //要操作的群组 (必填)
  "MemberToDel_Account": [ // 要删除的群成员列表, 最多500个
    "tommy",
    "jared"
  ]
}
```

- **静默删人**

当 Silence 为 1 时，成员成功删除后，不会给任何人下发系统通知。

```
{
  "GroupId": "@TGS#2J4SZEAE", //要操作的群组 (必填)
  "Silence": 1, // 是否静默删除 (选填)
  "MemberToDel_Account": [ // 要删除的群成员列表, 最多500个
    "tommy",
    "jared"
  ]
}
```

- **指定踢出原因**

指定踢出用户的原因，会在下发通知中体现。

```
{
  "GroupId": "@TGS#2J4SZEAE", //要操作的群组 (必填)
```

```

"Reason": "kick reason", //踢出用户原因 ( 选填 )
"MemberToDel_Account": [ // 要删除的群成员列表, 最多500个
"tommy",
"jared"
]
}
    
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。
Silence	Integer	选填	是否静默删人。0：非静默删人；1：静默删人。不填该字段默认为 0。
Reason	String	选填	踢出用户原因。
MemberToDel_Account	Array	必填	待删除的群成员。

应答

应答包体示例

```

{
"ActionStatus": "OK",
"ErrorInfo": "",
"ErrorCode": 0
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 <code>ErrorInfo</code> 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10005	请求包体中携带的用户数量过多（超过了 500 个成员），请减少一次请求的用户数量。
10007	操作权限不足。请确认操作者是否是 App 管理员，或被踢者是否为群主（部分群组不允许群主退出）。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 删除群组成员  
./TimRestApiGear.php group_open_http_svc delete_group_member (group_id) (member_id) (silence)
```

更多调试工具参见[REST API调试工具](#)。

API集成

PHP 集成

在云通信[PHP Server SDK](#)中，可以通过TimRestInterface的如下成员函数快速集成该API：

```
// 删除群组成员  
abstract function group_delete_group_member($group_id, $member_id, $silence);
```

可能触发的回调

群成员离开之后回调。

参考

REST API：增加群组成员（[v4/group_open_http_svc/add_group_member](#)）。

修改群成员资料

最近更新时间：2018-06-01 15:08:14

功能说明

App 管理员可以通过该接口修改群成员资料。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	支持修改管理员和群主资料
在线成员广播大群 (BChatRoom)	否

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室因为内部实现的问题，只能修改管理员和群主的成员资料，修改普通成员资料时会返回 10007 错误。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/modify_group_member_info?usersig=xxx&identifier=admin&sdkAppid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **设置管理员**

设置指定群成员的管理员身份。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "Role": "Admin" // 设置管理员
}
```

- **取消管理员**

取消指定群成员的管理员身份。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "Role": "Member" //取消管理员
}
```

- **设置成员消息屏蔽位**

设置的指定成员消息屏蔽类型：AcceptAndNotify 代表接收并提示消息，Discard 代表不接收也不提示消息，AcceptNotNotify 代表接收消息但不提示。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "MsgFlag": "AcceptAndNotify" // AcceptAndNotify、Discard或者AcceptNotNotify，消息屏蔽类型
}
```

• 设置成员的群名片

设置的指定成员的群名片，最大不超过50个字节。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "NameCard": "鲍勃" //群名片 (选填)
}
```

• 设置成员自定义字段

设置群成员的自定义字段。AppMemberDefinedData 默认是没有的，需要开通后才能使用，详见请求包字段说明表。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "AppMemberDefinedData": [ //要操作的成员自定义字段 (选填)
    {
      "Key": "MemberDefined1", //要操作的群成员自定义字段Key
      "Value": "ModifyData1" //要设置的数据内容
    },
    {
      "Key": "MemberDefined3",
      "Value": "ModifyData3"
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。
Member_Account	String	必填	要操作的群成员。
Role	String	选填	成员身份，Admin/Member分别为设置/取消管理员。
MsgFlag	String	选填	消息屏蔽类型。
NameCard	String	选填	群名片（最大不超过50个字节）。

字段	类型	属性	说明
AppMemberDefinedData	Array	选填	群成员维度的自定义字段，默认情况是没有的，需要开通，详情参见： 自定义字段 。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。

错误码	含义说明
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
80001	文本安全打击。请检查修改的群成员资料中是否包含敏感词汇。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 修改群组某成员身份
./TimRestApiGear.php group_open_http_svc modify_group_member_info (group_id) (account_id) \
(role)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 修改群成员身份
abstract function group_modify_group_member_info($group_id, $account_id, $role);
// 修改群成员资料(高级接口)
abstract function group_modify_group_member_info2($group_id, $account_id, $role,
$msg_flag, $shutup_time);
```

可能触发的回调

无。

参考

REST API : 获取群成员详细资料 ([v4/group_open_http_svc/get_group_member_info](#)) 。

解散群组

最近更新时间：2018-05-25 18:23:28

功能说明

App 管理员通过该接口解散群。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/destroy_group?usersig=xxx&identifier=admin&skappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#)提交工单申请处理。

请求

HTTP请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

群组解散之后将无法恢复，请谨慎调用该接口。

```
{
  "GroupId": "@TGS#2J4SZEAEEL"
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000 到 79999) 参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 解散群组
./TimRestApiGear.php group_open_http_svc destroy_group (group_id)
```

更多调试工具参见 [REST API调试工具](#)。

API集成

PHP集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 解散群
abstract function group_destroy_group($group_id);
```

可能触发的回调

群组解散之后回调。

参考

REST API : 创建群组 ([v4/group_open_http_svc/create_group](#)) 。

获取用户所加入的群组

最近更新时间：2018-05-25 18:24:44

功能说明

App 管理员可以通过该接口获取某一用户加入的群组。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群因为内部实现的差异，获取用户加入的群组时不会获取到这两种类型的群组。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_joined_group_list?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

• 基础形式

用来获取用户加入的群组，群组信息只包含所在群组的用户 ID。

```
{
  "Member_Account": "leckie"
}
```

• 分页拉取

可以使用 Limit 和 Offset 两个值用于控制分页拉取；

Limit 限制回包中 GroupIdList 数组中群组的个数，不得超过 5000；

Offset 控制从整个群组列表中的第多少个开始读取（默认从 0 开始）。对于分页请求（页码数字从 1 开始），每一页的 Offset 值应当为： $(\text{页码数} - 1) \times \text{每页展示的群组数量}$ ；

例如：假设需要分页拉取，每页展示 20 个，则第一页的请求参数应当为：`{ "Limit" : 20, "Offset" : 0}`，第二页的请求参数应当为 `{ "Limit" : 20, "Offset" : 20}`，依此类推；

Limit 或者 Offset 的取值不会对应答包体中的 TotalCount 造成影响。

```
{
  "Member_Account": "leckie",
  "Limit": 10, // 拉取多少个，不填标识拉取全部
  "Offset": 0 // 从第多少个开始拉取
}
```

• 指定群组形态

可以指定所拉取的群组所属的群组形态，如 Public，Private 和 ChatRoom（如果指定 AVChatRoom，获得的成员可能不完整）。

```
{
  "Member_Account": "leckie",
  "GroupType": "Public" // 拉取哪种群组形态，不填为拉取所有
}
```

- 拉取指定信息

可以指定拉取的基础信息字段，在 GroupBaseInfoFilter 中设置；

可以指定拉取的群成员自身在群内的信息，在 SelfInfoFilter 中设置。

```
{
  "Member_Account": "leckie",
  "ResponseFilter": {
    "GroupBaseInfoFilter": [ // 需要哪些基础信息字段
      "Type",
      "Name",
      "Introduction",
      "Notification",
      "FaceUrl",
      "CreateTime",
      "Owner_Account",
      "LastInfoTime",
      "LastMsgTime",
      "NextMsgSeq",
      "MemberNum",
      "MaxMemberNum",
      "ApplyJoinOption",
      "ShutUpAllMember"
    ],
    "SelfInfoFilter": [ // 需要自身在群内的信息
      "Role", // 群内身份
      "JoinTime", // 入群时间
      "MsgFlag", // 消息屏蔽类型
      "UnreadMsgNum" // 未读消息数量（IM云后台计算得出）
    ]
  }
}
```

- ALL IN ONE

```
{
  "Member_Account": "leckie",
  "Limit": 10, // 拉取多少个，不填标识拉取全部
  "Offset": 0, // 从第多少个开始拉取
  "GroupType": "Public", // 拉取哪种群组形态，不填为拉取所有
  "ResponseFilter": {
    "GroupBaseInfoFilter": [ // 需要哪些基础信息字段，含义参见“基本字段介绍”
      "Type",
      "Name",
      "Introduction",
      "Notification",
    ]
  }
}
```

```

"FaceUrl",
"CreateTime",
"Owner_Account",
"LastInfoTime",
"LastMsgTime",
"NextMsgSeq",
"MemberNum",
"MaxMemberNum",
"ApplyJoinOption",
"ShutUpAllMember"
],
"SelfInfoFilter": [ // 自身在群内的信息
"Role", // 群内身份
"JoinTime", // 入群时间
"MsgFlag", // 消息屏蔽类型
"UnreadMsgNum" // 未读消息数量 (IM云后台计算得出)
]
}
}
    
```

请求包字段说明

字段	类型	属性	说明
Member_Account	String	必填	需要查询的用户帐号。
Limit	Integer	选填	单次拉取的群组数量，如果不填代表所有群组，分页方式与 获取APP中的所有群组 相同。
Offset	Integer	选填	从第多少个群组开始拉取，分页方式与 获取APP中的所有群组 相同。
GroupType	String	选填	拉取哪种 群组形态 ，例如Private，Public，ChatRoom 或 AVChatRoom，不填为拉取所有。
ResponseFilter	Object	选填	分别包含 GroupBaseInfoFilter 和 SelfInfoFilter 两个过滤器；GroupBaseInfoFilter 表示需要拉取哪些基础信息字段，具体参见 群基础资料 ；SelfInfoFilter 表示需要拉取用户在每个群组中的哪些个人资料，具体请参见 群成员资料 。

应答

应答包体示例

- 基础形式和分页拉取

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "TotalCount": 2, // 不论Limit和Offset如何设置，该值总是满足条件的群组总数
  "GroupIdList": [
    {
      "GroupId": "@TGS#2J4SZEAEEL"
    },
    {
      "GroupId": "@TGS#2C5SZEAEF"
    }
  ]
}
    
```

- 指定群组形态

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "TotalCount": 1,
  "GroupIdList": [
    {
      "GroupId": "@TGS#2J4SZEAEEL"
    }
  ]
}
    
```

- 拉取指定信息

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "TotalCount": 2,
  "GroupIdList": [
    {
      "GroupId": "@TGS#2J4SZEAEEL",
      "Type": "Public",
      "LastMsgTime": 1425976500,
      "MemberCount": 48,
      "ShutUpAllMember": "On",
      "SelfInfo": {
        "Role": "Owner",
    
```

```

"MsgFlag": "AcceptAndNotify",
"UnreadMsgNum": 1
}
},
{
"GroupId": "@TGS#2C5SZEAEF",
"Type": "ChatRoom",
"LastMsgTime": 1425976567,
"MemberCount": 120,
"ShutUpAllMember": "Off",
"SelfInfo": {
"Role": "Member",
"MsgFlag": "AcceptAndNotify",
"UnreadMsgNum": 0
}
}
]
}
    
```

• ALL IN ONE

```

{
"ActionStatus": "OK",
"ErrorInfo": "",
"ErrorCode": 0,
"TotalCount": 1, // 不论Limit和Offset如何设置，该值总是满足条件的群组总数
"GroupIdList": [
{
"GroupId": "@TGS#2J4SZEAEEL",
"Type": "Public",
"LastMsgTime": 1425976500,
"MemberCount": 48,
"SelfInfo": {
"Role": "Owner",
"MsgFlag": "AcceptAndNotify",
"UnreadMsgNum": 1
}
}
]
}
    
```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
TotalCount	Integer	用户所加入的群组个数。
GroupIdList	Array	拉取到的群组信息，会根据高级接口中过滤器设置的过滤字段对结果进行结果进行过滤。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10018	应答包长度超限。因为请求的内容过多，导致应答包超过了最大包长（1MB），请尝试减少单次请求的数据量。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取用户所加入的所有群组
./TimRestApiGear.php group_open_http_svc get_joined_group_list (account_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 获取某一用户加入的群组  
abstract function group_get_joined_group_list($account_id);  
  
// 获取某一用户加入的群组(高级接口)  
abstract function group_get_joined_group_list2($account_id, $group_type,  
$base_info_filter, $self_info_filter);
```

可能触发的回调

无。

参考

无。

查询用户在群组中的身份

最近更新时间：2018-05-25 18:25:57

功能说明

App 管理员可以通过该接口获取一批用户在群内的身份。

接口调用说明

适用的群组类型

群组类型	支持此REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明1)
在线成员广播大群 (BChatRoom)	否 (见说明2)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：

- 互动直播聊天室不支持使用该 REST API 查询用户在群组中的身份，如果对这种群组进行操作将返回 10007 错误。如果管理员希望达到查询用户身份的效果，可以通过 [获取群组成员详细资料](#) 实现。
- 在线成员广播大群不支持设置管理员和群主，全部成员都为普通成员。所以也不支持使用该 REST API 查询用户在群组中的身份。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_role_in_group?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

用来获取用户加入的群组，群组信息包含群 ID 和需要查询身份的群成员 ID。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "User_Account": [ // 最多支持500个
    "leckie",
    "peter",
    "wesley"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	需要查询的群组 ID。
User_Account	Array	必填	表示需要查询的用户帐号，最多支持 500 个帐号。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "UserIdList": [ // 结果
    {
```

```

"Member_Account": "leckie",
"Role": "Owner" // 身份 : Owner/Admin/Member/NotMember
},
{
"Member_Account": "peter",
"Role": "Member"
},
{
"Member_Account": "wesley",
"Role": "NotMember"
}
]
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
UserIdList	Array	拉取到的成员在群内的身份信息，可能的身份包括 Owner/Admin/Member/NotMember 。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。

错误码	含义说明
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 获取用户所加入的所有群组
./TimRestApiGear.php group_open_http_svc get_joined_group_list (account_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 查询用户在某个群组中的身份
abstract function group_get_role_in_group($group_id, $member_id);
```

可能触发的回调

无。

参考

无。

批量禁言和取消禁言

最近更新时间：2018-05-25 18:26:14

功能说明

App 管理员禁止指定群组中某些用户在一段时间内发言；

App 管理员取消对某些用户的禁言；

被禁言用户退出群组之后再进入同一群组，禁言仍然有效。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	否 (见说明)
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：1.私有群不支持禁言；2.在线成员广播大群只有 APP 管理员可以发送消息，所以无需支持设置和取消禁言。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/forbid_send_msg?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **设置禁言**

可以通过 ShutUpTime 设置禁言时间，对指定用户禁言。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "Members_Account": [ // 最多支持500个
    "peter",
    "leckie"
  ],
  "ShutUpTime": 60 // 禁言时间，单位为秒
}
```

- **取消禁言**

如果要取消禁言，将 ShutUpTime 设为 0。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "Members_Account": [ // 最多支持 500 个
    "peter",
    "leckie"
  ],
  "ShutUpTime": 0 // 为 0 时表示取消禁言
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	需要查询的群组 ID。
Members_Account	Array	必填	需要禁言的用户帐号，最多支持 500 个帐号。

字段	类型	属性	说明
ShutUpTime	Integer	必填	需禁言时间，单位为秒，为 0 时表示取消禁言。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组ID非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 在群组中禁言用户  
./TimRestApiGear.php group_open_http_svc forbid_send_msg (group_id) (member_id) (second)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 在群组中禁言用户  
abstract function group_forbid_send_msg($group_id, $member_id, $second);
```

可能触发的回调

无。

参考

REST API：获取群组被禁言用户列表 ([v4/group_open_http_svc/get_group_shutted_uin](#))。

获取群组被禁言用户列表

最近更新时间：2018-05-25 18:26:30

功能说明

App 管理员可以根据群组 ID 获取群组中被禁言的用户列表。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	否

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/get_group_shutted_uin?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP请求包体格式

JSON

请求包示例

用来获取群组中被禁言的成员列表，只包含群ID。

```
{
  "GroupId": "@TGS#1KGZ2RAEU"
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	需要获取被禁言成员列表的群组 ID。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "GroupId": "@TGS#2FZNNRAEU",
  "ShuttedUinList": [ //群组中被禁言的用户列表
    {
      "Member_Account": "tommy", //用户ID
      "ShuttedUntil": 1458115189 //禁言到的时间（使用UTC时间，即世界协调时间）
    },
    {
      "Member_Account": "peter",
      "ShuttedUntil": 1458115189
    }
  ]
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。

字段	类型	说明
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
ShuttedUinList	Array	返回结果为禁言用户信息数组，内容包括被禁言的成员ID，及其被禁言到的时间（使用UTC时间，即世界协调时间）。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是APP管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API：批量禁言和取消禁言 ([v4/group_open_http_svc/forbid_send_msg](#))。

在群组中发送普通消息

最近更新时间：2018-05-25 18:26:46

功能说明

App 管理员可以通过该接口在群组中发送普通消息。

接口调用说明

适用的群组类型

群组类型	支持此REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	是
在线成员广播大群 (BChatRoom)	是

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/send_group_msg?usersig=xxx&identifier=admin  
&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例**• 基础形式**

发送普通群消息，消息的发送者为 App 管理员。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "Random": 8912345, // 随机数字，五分钟数字相同认为是重复消息
  "MsgBody": [ // 消息体，由一个element数组组成，详见字段说明
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    },
    {
      "MsgType": "TIMFaceElem", // 表情
      "MsgContent": {
        "Index": 6,
        "Data": "abc\u0000\u0001"
      }
    }
  ],
  "OfflinePushInfo": {
    "PushFlag": 0, // 正常推送，
    "Desc": "离线推送内容",
    "Ext": "这是透传的内容",
    "AndroidInfo": {
      "Sound": "android.mp3"
    },
    "ApnsInfo": {
      "Sound": "apns.mp3",
      "BadgeMode": 1, // 这个字段缺省或者为 0 表示需要计数，为 1 表示本条消息不需要计数，即右上角图标数字不增加
      "Title": "apns title", // apns title
      "SubTitle": "apns subtitle", // apns subtitle
      "Image": "www.image.com" // image url
    }
  }
}
```

• 指定消息发送者

可以指定某一位群成员作为消息发送者，在 From_Account 中设置；
其他用户收到群消息后，显示的消息来自 App 管理员指定的那一位群成员。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "From_Account": "leckie", //指定消息发送者（选填）
  "Random": 8912345, //随机数字，五分钟数字相同认为是重复消息
  "MsgBody": [ //消息体，由一个element数组组成，详见字段说明
    {
      "MsgType": "TIMTextElem", //文本
      "MsgContent": {
        "Text": "red packet"
      }
    },
    {
      "MsgType": "TIMFaceElem", //表情
      "MsgContent": {
        "Index": 6,
        "Data": "abc\u0000\u0001"
      }
    }
  ]
}
```

• 指定消息优先级

可以指定消息的优先级，默认优先级 Normal；
可以指定4种优先级，从高到低依次为 High，Normal，Low，Lowest，区分大小写。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "Random": 8912345, //随机数字，五分钟数字相同认为是重复消息
  "MsgPriority": "High", //消息优先级
  "MsgBody": [ //消息体，由一个element数组组成，详见字段说明
    {
      "MsgType": "TIMTextElem", //文本
      "MsgContent": {
        "Text": "red packet"
      }
    },
    {
      "MsgType": "TIMFaceElem", //表情
      "MsgContent": {
        "Index": 6,
        "Data": "abc\u0000\u0001"
      }
    }
  ]
}
```

```
}  
}  
]  
}
```

- **指定消息不回调**

在回调开关打开的情况下，用户通过 ForbidCallbackControl 控制单条消息是否发起回调，默认情况发起回调。

```
{  
  "GroupId": "@TGS#2C5SZEAEF",  
  "Random": 8912345, // 随机数字，五分钟数字相同认为是重复消息  
  "ForbidCallbackControl": [  
    "ForbidBeforeSendMsgCallback",  
    "ForbidAfterSendMsgCallback"], // 禁止回调控制选项  
  "MsgBody": [ // 消息体，由一个element数组组成，详见字段说明  
    {  
      "MsgType": "TIMTextElem", // 文本  
      "MsgContent": {  
        "Text": "red packet"  
      }  
    },  
    {  
      "MsgType": "TIMFaceElem", // 表情  
      "MsgContent": {  
        "Index": 6,  
        "Data": "abc\u0000\u0001"  
      }  
    }  
  ]  
}
```

- **指定消息不存离线及漫游**

如果消息体中指定 OnlineOnlyFlag，只要值大于 0，则消息表示只在线下发，不存离线和漫游（AVChatRoom 和 BChatRoom 不允许使用）。

```
{  
  "GroupId": "@TGS#2C5SZEAEF",  
  "Random": 8912345, // 随机数字，五分钟数字相同认为是重复消息  
  "OnlineOnlyFlag": 1, // 表示只在线下发(只有在线群成员才能收到)，不存离线及漫游  
  "MsgBody": [ // 消息体，由一个 element 数组组成，详见字段说明  
    {  
      "MsgType": "TIMTextElem", // 文本  
      "MsgContent": {  
        "Text": "red packet"  
      }  
    }  
  ]  
}
```

```

    },
    {
      "MsgType": "TIMFaceElem", // 表情
      "MsgContent": {
        "Index": 6,
        "Data": "abc\u0000\u0001"
      }
    }
  ]
}
    
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	向哪个群组发送消息。
Random	Integer	必填	32 位随机数。如果 5 分钟内两条消息的随机值相同，后一条消息将被当做重复消息而丢弃。
MsgPriority	String	选填	消息的优先级。
MsgBody	Array	必填	消息体，具体参见 消息格式描述 。
From_Account	String	选填	消息来源帐号，选填。如果不填写该字段，则默认消息的发送者为调用该接口时使用的 App 管理员 帐号。除此之外，App 亦可通过该字段“伪造”消息的发送者，从而实现一些特殊的功能需求。需要注意的是，如果指定该字段，必须要确保字段中的帐号是存在的。
OfflinePushInfo	Object	选填	离线推送信息配置，具体可参考 消息格式描述 。
ForbidCallbackControl	Array	选填	消息回调禁止开关，只对单条消息有效， ForbidBeforeSendMsgCallback 表示禁止发消息前回调， ForbidAfterSendMsgCallback 表示禁止发消息后回调。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MsgTime": 1497249503,
}
    
```

```
"MsgSeq": 1
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
MsgTime	Integer	消息发送的时间戳，对应后台 server 时间。
MsgSeq	Integer	消息序列号，唯一标示一条消息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10016	该请求触发了到 App 后台的“ 群内发言之前回调 ”，云通讯后台根据回调结果拒绝下发该消息。
10017	因被禁言而不能发送消息，请检查发送者是否被设置禁言。
10023	请求命令非法，请再次尝试或联系技术客服。
80001	文本安全打击。请检查消息文本中是否有敏感词汇。
80002	消息内容过长，目前最大支持 8000 字节的消息，请调整消息长度。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 群组中发送普通消息
./TimRestApiGear.php group_open_http_svc send_group_msg (account_id) (group_id) \
(text_content)

# 群组中发送图片
./TimRestApiGear.php group_open_http_svc send_group_msg_pic (account_id) (group_id) \
(pic_path)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 在某一群组里发普通消息
abstract function group_send_group_msg($account_id, $group_id, $text_content);

// 在某一群组里发送图片
abstract function group_send_group_msg_pic($account_id, $group_id, $pic_path);

// 在某一群组里发普通消息(高级接口)
abstract function openim_send_msg2($account_id, $receiver, $msg_content);
```

可能触发的回调

[群内发言之前回调](#)。

参考

REST API：在群组中发送系统通知（[v4/group_open_http_svc/send_group_system_notification](#)）；

REST API：单发单聊消息（[v4/openim/sendmsg](#)）；

REST API：批量发单聊消息（[v4/openim/batchsendmsg](#)）；

[消息格式描述](#)；

[群组内普通消息与系统通知之间的区别](#)。

在群组中发送系统通知

最近更新时间：2018-05-25 18:27:00

功能说明

App 管理员可以通过该接口在群组中发送系统通知。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是 (可指定成员)
公开群 (Public)	是 (可指定成员)
聊天室 (ChatRoom)	是 (可指定成员)
互动直播聊天室 (AVChatRoom)	是 (全员)
在线成员广播大群 (BChatRoom)	是 (全员)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：私有群、公开群、聊天室支持向群组中的一部分指定成员发送系统通知，而互动直播聊天室和在线成员广播大群只支持向群组中所有成员发送系统通知。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/send_group_system_notification?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见[REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **基础形式**

用来向群中的所有群成员下发系统消息。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "Content": "Hello World" // 系统通知内容
}
```

- **指定接收群成员**

用来向群中的指定的群成员下发系统消息，接收者在 ToMembers_Account 中设置（互动直播聊天室不支持指定接收群成员）。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "ToMembers_Account": [ // 接收者群成员列表，不填或为空表示全员下发
    "peter",
    "leckie"
  ],
  "Content": "Hello World" // 系统通知内容
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	向哪个群组发送系统通知。
ToMembers_Account	String	选填	接收者群成员列表，不填或为空表示全员下发。
Content	String	必填	系统通知的内容。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

可以通过 [PHP Server SDK](#) 中的调试工具进行简单的 REST API 调试，命令如下：

```
# 群组中发送系统消息
./TimRestApiGear.php group_open_http_svc send_group_system_notification (group_id) \
(text_content) (receive_id)
```

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

在云通信 [PHP Server SDK](#) 中，可以通过 TimRestInterface 的如下成员函数快速集成该 API：

```
// 在某一群组发系统消息
abstract function group_send_group_system_notification($group_id, $content, $receiver_id);

// 在某一群组发系统消息(高级接口)
abstract function group_send_group_system_notification2($group_id, $content, $receiver_list);
```

可能触发的回调

无。

参考

REST API：在某一群组中发送普通消息（[v4/group_open_http_svc/send_group_msg](#)）；
[群组内普通消息与系统通知之间的区别](#)。

群组消息撤回

最近更新时间：2018-07-25 16:49:44

功能说明

App 管理员通过该接口撤回指定群组的消息，消息需要在漫游有效期以内。

接口调用说明

适用的群组类型

群组类型	支持群消息撤回
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否
在线成员广播大群 (BChatRoom)	否

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/group_msg_recall?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

消息撤回之后将无法恢复，请谨慎调用该接口。

```
{
  "GroupId": "@TGS#2J4SZEAEI",
  "MsgSeqList": [
    {
      "MsgSeq": 100
    },
    {
      "MsgSeq": 101
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。
MsgSeqList	Array	必填	被撤回的消息 seq 列表，一次请求最多可以撤回 10 条消息 seq。
MsgSeq	Integer	必填	请求撤回的消息 seq

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "RecallRetList": [
    {
      "MsgSeq": 100,
      "RetCode": 10030
    },
    {

```

```
"MsgSeq":101,
"RetCode":0
}
]
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
RecallRetList	Array	消息撤回请求的详细结果。
MsgSeq	String	单个被撤回消息的 seq。
RetCode	String	单个消息的被撤回结果：0 表示成功；其它表示失败，参考下文错误码说明。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10030	请求撤回的消息不存在。
10031	请求撤回的消息超出时间限制。

错误码	含义说明
10032	请求撤回的消息不支持撤回操作。
10033	群组类型不支持消息撤回操作。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

转让群组

最近更新时间：2018-05-25 18:27:43

功能说明

App 管理员可以通过该接口将群主转移给他人；
没有群主的群也可以转让，新群主不能为空。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不支持转让群组，对这两种类型的群组进行操作时会返回 10007 错误。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/change_group_owner?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

转移群组给其他人，转移的新群主必须是群成员。

```
{
  "GroupId": "@TGS#1NVTZEAE4", // 要被转移的群ID (必填)
  "NewOwner_Account": "peter", // 新群主ID (必填)
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	要被转移的群组 ID。
NewOwner_Account	String	必填	新群主 ID。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。

字段	类型	说明
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

无。

导入群基础资料

最近更新时间：2018-07-25 18:12:49

功能说明

App 管理员可以通过该接口导入群组，不会触发回调、不会下发通知。当 App 需要从其他即时通信系统迁移到云通信时，使用该协议导入存量群组数据。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不支持导入群基础资料，对这两种类型的群组进行操作时会返回 10007 错误。如果需要达到导入群基础资料的效果，可以通过 [创建群组](#) 和 [修改群组基础资料](#) 的方式实现。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/import_group?usersig=xxx&identifier=admin&skappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **基础形式**

导入群，可以用 CreateTime 指定群组的创建时间。

```
{
  "Owner_Account": "leckie", // 群主的UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom (必填)
  "Name": "TestGroup", // 群名称 (必填)
  "CreateTime": 1448357837 // 群组的创建时间 (选填, 不填会以请求时刻为准)
}
```

- **指定其他选填字段**

指定简介、公告等选填字段，和创建群组相同。

```
{
  "Owner_Account": "leckie", // 群主的 UserId (选填)
  "Type": "Public", // 群组类型: Private/Public/ChatRoom (必填)
  "GroupId": "MyFirstGroup", // 用户自定义群组外显 ID (选填)
  "Name": "TestGroup", // 群名称 (必填)
  "Introduction": "This is group Introduction", // 群简介 (选填)
  "Notification": "This is group Notification", // 群公告 (选填)
  "FaceUrl": "http://this.is.face.url",
  "MaxMemberCount": 500, // 最大群成员数量 (选填)
  "ApplyJoinOption": "FreeAccess", // 申请加群处理方式 (选填)
  "CreateTime": 1448357837, // 群组的创建时间 (选填, 不填会以请求时刻为准)
  "AppDefinedData": [ // 群组维度的自定义字段 (选填)
    {
      "Key": "GroupTestData1", // App 自定义的字段 Key
      "Value": "xxxxx" // 自定义字段的值
    },
    {
      "Key": " GroupTestData2",
      "Value": "abc\u0000\u0001" // 自定义字段支持二进制数据
    }
  ]
}
```

```
}
]
}
```

请求包字段说明

字段	类型	属性	说明
Owner_Account	String	选填	群主 ID，自动添加到群成员中。如果不填，群没有群主。
Type	String	必填	群组形态 ，包括 Public（公开群），Private（私密群），ChatRoom（聊天室）。
GroupId	String	选填	为了使得群组ID更加简单，便于记忆传播，腾讯云支持APP在通过REST API创建群组时自定义群组ID。详情参见： 自定义群组ID 。
Name	String	必填	群名称，最长 30 字节。
Introduction	String	选填	群简介，最长 240 字节。
Notification	String	选填	群公告，最长 300 字节。
FaceUrl	String	选填	群头像URL，最长 100 字节。
MaxMemberCount	Integer	选填	最大群成员数量，最大为 10000，不填默认为 2000个。
ApplyJoinOption	String	选填	申请加群处理方式。包含 FreeAccess（自由加入），NeedPermission（需要验证），DisableApply（禁止加群），不填默认为NeedPermission（需要验证）。
AppDefinedData	Array	选填	群组维度的自定义字段，默认情况是没有的，需要开通，详情参见： 自定义字段 。
CreateTime	Integer	选填	群组的创建时间。

应答

应答包体示例

- 基础形式

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
```

```
"GroupId": "@TGS#2J4SZEDEL"
}
```

- 指定其他选填字段

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "GroupId": "MyFirstGroup"
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
GroupId	String	创建成功之后的群ID，由IM云后台分配或者用户指定。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10021	群组 ID 已被使用，请选择其他的群组 ID。
80001	文本安全打击。请检查群名称、群公告和群简介等是否有敏感词汇。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API：设置成员未读计数（[v4/group_open_http_svc/set_unread_msg_num](#)）。

REST API：导入群成员（[v4/group_open_http_svc/import_group_member](#)）。

REST API：解散群组（[v4/group_open_http_svc/destroy_group](#)）。

导入群消息

最近更新时间：2018-05-25 18:28:15

功能说明

该 API 接口的作用是导入群组的消息，不会触发回调、不会下发通知。

当 App 需要从其他即时通信系统迁移到云通信时，使用该协议导入存量群消息数据。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不支持导入群消息，对这两种类型的群组进行操作时会返回 10007 错误。因为这两种类型群组所适用的场景一般不需要历史消息，所以没有提供这一功能。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/import_group_msg?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

批量导入群消息，一次最多导入 20 条；

使用本接口导入消息后所有成员的未读计数都会变成 0，如果要保留未读计数，请在导入所有消息后再 [导入群成员](#) 或者 [设置成员未读计数](#)；

导入的消息必须按时间戳的顺序导入，且时间戳必须大于建群时间且小于当前时间，否则会失败。

```
{
  "GroupId": "@TGS#2C5SZEAEF",
  "MsgList": [
    {
      "From_Account": "leckie", //指定消息发送者
      "SendTime": 1448357837,
      "Random": 8912345, // 消息随机数 (可选)
      "MsgBody": [ // 消息体，由一个 element 数组组成，详见 TIMMessage 消息对象
        {
          "MsgType": "TIMTextElem", // 文本
          "MsgContent": {
            "Text": "red packet"
          }
        },
        {
          "MsgType": "TIMFaceElem", // 表情
          "MsgContent": {
            "Index": 6,
            "Data": "abc\u0000\u0001"
          }
        }
      ]
    },
    {
      "From_Account": "peter", //指定消息发送者
      "SendTime": 1448357837,
```

```

"MsgBody": [ // 消息体，由一个 element 数组组成，详见 TIMMessage 消息对象
{
"MsgType": "TIMTextElem", // 文本
"MsgContent": {
"Text": "red packet"
}
}
]
}

]
}
    
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	要导入消息的群 ID。
MsgList	String	必填	导入的消息列表。
From_Account	String	必填	指定消息发送者。
SendTime	Integer	必填	指定消息发送者。
Random	Integer	选填	32 位随机数。如果 5 分钟内两条消息的随机值相同，后一条消息将被当做重复消息而丢弃。
MsgBody	Object	必填	TIM 消息，请参考 TIMMsgElement对象的定义 。
MsgType	String	必填	TIM 消息对象类型，目前支持的消息对象包括：TIMTextElem(文本消息)，TIMFaceElem(表情消息)，TIMLocationElem(位置消息)，TIMCustomElem(自定义消息)。
MsgContent	Object	必填	MsgContent 为 TIM 消息对象，具体可参考 TIMMsgElement对象的定义 。

应答

应答包体示例

```

{
"ActionStatus": "OK",
"ErrorCode": 0,
"ImportMsgResult": [
{
    
```

```

"Result": 0,
"MsgSeq": 1,
"MsgTime": 1449454106
},
{
"Result": 0,
"MsgSeq": 2,
"MsgTime": 1449454116
},
{
"Result": 0,
"MsgSeq": 3,
"MsgTime": 1449454106
},
{
"Result": 0,
"MsgSeq": 4,
"MsgTime": 1449454116
}
]
}
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ImportMsgResult	Array	具体的消息导入结果。
Result	Integer	单条消息导入结果。
MsgSeq	Integer	给消息分配的 seq。
MsgTime	Integer	消息的时间戳。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 <code>ErrorInfo</code> 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	操群组 ID 非法，请检查群组 ID 是否填写正确。
10020	消息内容过长，目前最大支持 8000 字节的消息，请调整消息长度。
80001	文本安全打击。请检查群名称、群公告和群简介等是否有敏感词汇。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API：设置成员未读消息计数（[v4/group_open_http_svc/set_unread_msg_num](#)）。

导入群成员

最近更新时间：2018-05-25 18:28:31

功能说明

该 API 接口的作用是导入群组成员，不会触发回调、不会下发通知。

当 App 需要从其他即时通信系统迁移到云通信时，使用该协议导入存量群成员数据。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不支持导入群成员，对这两种类型的群组进行操作时会返回 10007 错误。因为这两种类型群组所适用的场景一般不需要导入成员，所以没有提供这一功能。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/import_group_member?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

用来向群中导入成员，一次请求最多支持添加 500 个成员；使用本接口设置的未读消息数如果大于群当前的消息数，未读消息数会设为群当前的消息总数；

请保证导入成员的入群时间大于群的创建时间并小于当前时间，否则该成员会导入失败。

```
{
  "GroupId": "@TGS#2J4SZEAE", // 要操作的群组 (必填)
  "MemberList": [ // 一次最多添加 500 个成员
    {
      "Member_Account": "tommy" // 要添加的群成员 ID (必填)
      "Role": "Admin", // 导入成员的角色, 目前只有 Admin(可选)
      "JoinTime": 1448357837, // 导入的成员入群时间 (选填)
      "UnreadMsgNum": 5 // 该成员的未读消息数 (选填)
    },
    {
      "Member_Account": "jared"
      "JoinTime": 1448357857,
      "UnreadMsgNum": 2
    }
  ]
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。

字段	类型	属性	说明
MemberList	Array	必填	待添加的群成员数组。
Member_Account	String	必填	待导入的群成员帐号。
Role	String	选填	待导入群成员角色。目前只支持填 Admin，不填则为普通成员 Member。
JoinTime	Integer	选填	待导入群成员的入群时间。
UnreadMsgNum	Integer	选填	待导入群成员的未读消息计数。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "MemberList": [
    {
      "Member_Account": "tommy",
      "Result": 1 // 导入结果：0为失败；1为成功；2表示已经是群成员
    },
    {
      "Member_Account": "jared",
      "Result": 1
    }
  ]
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
MemberList	Array	返回添加的群成员结果。
Member_Account	String	返回的群成员帐号。

字段	类型	说明
Result	Integer	导入结果：0 为失败；1 为成功；2 表示已经是群成员。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10005	请求包体中携带的用户数量过多（超过了 500 个成员），请减少一次请求的用户数量。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。
10019	被添加用户的帐号不存在，请检查用户帐号是否正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API：删除群组成员 ([v4/group_open_http_svc/delete_group_member](#)) 。

设置成员未读消息计数

最近更新时间：2018-05-25 18:35:12

功能说明

App 管理员使用该接口设置群组成员未读消息数，不会触发回调、不会下发通知。

当 App 需要从其他即时通信系统迁移到云通信时，使用该协议设置群成员的未读消息计数。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	否 (见说明)
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：聊天室、互动直播聊天室和在线成员广播大群的客户端不支持未读消息计数，所以对这三种类型的群组成员设置未读消息计数是无效的（但是不会返回错误）。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/set_unread_msg_num?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

设置指定群成员的未读消息计数；

使用本接口设置的未读计数如果大于群当前的消息数，未读计数会设为群当前的消息总数。

```
{
  "GroupId": "@TGS#2CLUZEAEJ", //要操作的群组 (必填)
  "Member_Account": "bob", // 要操作的群成员 (必填)
  "UnreadMsgNum":5 // 该成员的未读消息数
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	操作的群 ID。
Member_Account	String	必填	要操作的群成员。
UnreadMsgNum	Integer	必填	成员未读消息数。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

无。

删除指定用户发送的消息

最近更新时间：2018-05-25 18:29:22

功能说明

该 API 接口的作用是删除最近 1000 条消息中指定用户发送的消息。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不会存储漫游消息（这里的删除消息即删除漫游消息），对这两种类型的群组中的用户删除消息是无效的（但是不会返回错误）。而且在在线成员广播大群中，只允许 App 管理员发送消息，普通用户无法发送消息。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/delete_group_msg_by_sender?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

删除群消息，删除最近 1000 条消息内某个人发送的消息。

```
{
  "GroupId": "@TGS#2C5SZEAEF", //必填
  "Sender_Account": "leckie" //必填
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	要删除消息的群 ID。
Sender_Account	String	必填	被删除消息的发送者 ID。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。

字段	类型	说明
ErrorCode	Integer	错误码。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	无效的命令字或命令号。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API集成

无。

可能触发的回调

无。

参考

无。

拉取群漫游消息

最近更新时间：2018-05-25 18:29:37

功能说明

App 管理员可以通过该接口拉取群组的漫游消息。

背景说明

1. IM 云的消息是按 Seq 排序的，按照 server 收到消息的顺序分配 Seq，先发的消息 Seq 小，后发的 Seq 大。
2. 如果用户想拉取一个群的全量消息，首次拉取时不用填拉取 Seq，Server 会自动返回最新的消息，以后拉取时拉取 Seq 填上次返回的最小 Seq 减 1。
3. 如果返回消息的 IsPlaceMsg 为 1，表示这个 Seq 的消息或者过期、或者存储失败、或者被删除了。

接口调用说明

适用的群组类型

群组类型	支持此 REST API
私有群 (Private)	是
公开群 (Public)	是
聊天室 (ChatRoom)	是
互动直播聊天室 (AVChatRoom)	否 (见说明)
在线成员广播大群 (BChatRoom)	否 (见说明)

云通信中内置以上五种群组类型，详情请见 [群组形态介绍](#)。

说明：互动直播聊天室和在线成员广播大群不存储漫游消息。

请求 URL

```
https://console.tim.qq.com/v4/group_open_http_svc/group_msg_get_simple?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

- **基础形式**

拉取群组的漫游消息，返回群组最新的 ReqMsgNumber 条消息。

```
{
  "GroupId": "@TGS#15ERQPAER", //拉取消息的群ID
  "ReqMsgNumber": 2 //需要拉取的消息条数
}
```

- **按 seq 拉取**

按指定 seq 拉取群组的漫游消息；

返回消息 seq 小于等于 ReqMsgSeq 的 ReqMsgNumber 条消息。

```
{
  "GroupId": "@TGS#15ERQPAER",
  "ReqMsgSeq": 7803321, //请求的消息最大seq，返回<=ReqMsgSeq的消息
  "ReqMsgNumber": 2
}
```

请求包字段说明

字段	类型	属性	说明
GroupId	String	必填	要拉取漫游消息的群组 Id。
ReqMsgNumber	Integer	必填	拉取的漫游消息的条数，目前一次请求最多返回 20 条漫游消息，所以这里最好小于等于 20。
ReqMsgSeq	Integer	选填	拉取消息的最大 seq。

应答

应答包体示例

```

{
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "GroupId": "@TGS#15ERQPAER",
  "IsFinished": 1,
  "RspMsgList": [
    {
      "From_Account": "144115197276518801",
      "IsPlaceMsg": 0,
      "MsgBody": [
        {
          "MsgContent": {
            "Data": "\b\u0001\u0010\u0006\u001A\u0006猫瞳",
            "Desc": "MIF",
            "Ext": ""
          },
          "MsgType": "TIMCustomElem"
        },
        {
          "MsgContent": {
            "Data": "",
            "Index": 15
          },
          "MsgType": "TIMFaceElem"
        }
      ],
      "MsgRandom": 51083293,
      "MsgSeq": 7803321,
      "MsgTimeStamp": 1458721802
    },
    {
      "From_Account": "144115198339527735",
    }
  ]
}
    
```

```

    "IsPlaceMsg": 0,
    "MsgBody": [
    {
    "MsgContent": {
    "Data": "\b\u0001\u0010\u0006\u001A\u000F西瓜妹妹。",
    "Desc": "MIF",
    "Ext": ""
    },
    "MsgType": "TIMCustomElem"
    },
    {
    "MsgContent": {
    "Text": "报上来"
    },
    "MsgType": "TIMTextElem"
    }
    ],
    "MsgRandom": 235168582,
    "MsgSeq": 7803320,
    "MsgTimeStamp": 1458721797
    }
    ]
    }
    
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，OK 表示处理成功，FAIL 表示失败。
ErrorCode	Integer	错误码。
GroupId	String	请求中的群组Id。
IsFinished	Integer	是否返回了请求区间的全部消息，当消息长度太长或者区间太大（超过 20）导致无法返回全部消息时值为 0。
RspMsgList	Array	返回的消息列表。
From_Account	String	消息的发送者。
IsPlaceMsg	Integer	是否是空洞消息，当消息被删除或者消息过期后，MsgBody 为空，这个字段为 1。
MsgRandom	Integer	消息随机值，用来对消息去重，有客户端发消息时填写，如果没有填，服务端会自动生成一个。

字段	类型	说明
MsgSeq	Integer	消息 seq，用来标识唯一消息，值越小发送的越早。
MsgTimeStamp	Integer	消息被发送的时间戳，server 的时间。
MsgBody	Array	消息内容。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员。
10010	群组不存在，或者曾经存在过，但是目前已经被解散。
10015	群组 ID 非法，请检查群组 ID 是否填写正确。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

无。

脏字管理

查询APP自定义脏字

最近更新时间：2018-05-25 18:36:24

背景说明

1. 脏字指云通信为 App 配置的敏感词，如果检查的内容包含脏字，将拒绝请求，并返回 [80001错误](#)；
2. 脏字检查的内容包括群组消息（只检查文本消息 [TIMTextElem](#)，不支持对自定义消息 [TIMCustomElem](#) 的过滤）、群组资料（群名称、群简介、群公告）和群名片；
3. 云通信的脏字集中已经涵盖了一批默认脏字（政治、色情等领域）；
4. 如果App有除默认脏字集之外的自定义脏字需求，可以通过脏字管理的REST API进行配置。

功能说明

App 管理员可以通过该接口查询已设置的自定义脏字。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim_dirty_words/get?usersig=xxx&identifier=admin&sdkappid=8888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

查询已经对 App 设置的自定义脏字。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "DirtyWordsList": [
    "回收金币",
    "代刷人气",
    "抽奖"
  ]
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
DirtyWordsList	Array	返回结果是已经设置的自定义脏字。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10007	操作权限不足。请确认操作者是否是 APP管理员 。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API : 添加 App 的自定义脏字 ([v4/openim_dirty_words/add](#)) ;

REST API : 删除 App 的自定义脏字 ([v4/openim_dirty_words/delete](#)) 。

添加APP自定义脏字

最近更新时间：2018-06-14 17:59:40

背景说明

1. 脏字指云通信为 App 配置的敏感词，如果检查的内容包含脏字，将拒绝请求，并返回 [80001错误](#)；
2. 脏字检查的内容包括群组消息（只检查文本消息 [TIMTextElem](#)，不支持对自定义消息 [TIMCustomElem](#) 的过滤）、群组资料（群名称、群简介、群公告）和群名片；
3. 云通信的脏字集中已经涵盖了一批默认脏字（政治、色情等领域）；
4. 如果 App 有除默认脏字集之外的自定义脏字需求，可以通过脏字管理的 REST API 进行配置。

功能说明

1. App 管理员可以通过该接口添加自定义的脏字；
2. 添加自定义脏字成功后，需要等待 5 分钟才能生效；
3. 默认 IM 添加脏字不能超过 50 个，如果超过 50 个则多次请求即可。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openim_dirty_words/add?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

支持批量添加 App 自定义的脏字，添加的新脏字和已有的脏字数量之和不超过 50 个，，如果超过 50 个则需要多次请求。

```
{
  "DirtyWordsList": [ // 自定义脏字列表（必填），列表中的脏字不能超过 50 个
    "韩国代购", //每个自定义脏字不能超过 200 字节
    "发票"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
DirtyWordsList	Array	必填	该字段用来指定需要添加的自定义脏字。添加的新脏字和已有的脏字数量之和不超过50个，每个脏字的长度不能超过200字节。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 ErrorInfo 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 APP管理员 。

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API : 查询 App 的自定义脏字 ([v4/openim_dirty_words/get](#)) ;

REST API : 删除 App 的自定义脏字 ([v4/openim_dirty_words/delete](#)) 。

删除APP自定义脏字

最近更新时间：2018-05-25 18:32:24

背景说明

1. 脏字指云通信为App配置的敏感词，如果检查的内容包含脏字，将拒绝请求，并返回 **80001错误**；
2. 脏字检查的内容包括群组消息（只检查文本消息 **TIMTextElem**，不支持对自定义消息 **TIMCustomElem** 的过滤）、群组资料（群名称、群简介、群公告）和群名片；
3. 云通信的脏字集中已经涵盖了一批默认脏字（政治、色情等领域）；
4. 如果 App 有除默认脏字集之外的自定义脏字需求，可以通过脏字管理的 REST API 进行配置。

功能说明

App 管理员可以通过该接口删除自定义的脏字。

接口调用说明

请求URL

```
https://console.tim.qq.com/v4/openim_dirty_words/delete?usersig=xxxidentifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

支持批量删除 App 自定义的脏字。

```
{
  "DirtyWordsList": [ // 自定义脏字列表（必填），列表中的脏字不能超过 50 个
    "韩国代购", //每个自定义脏字不能超过 200 字节
    "发票"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
DirtyWordsList	Array	必填	该字段用来指定需要添加的自定义脏字。添加的新脏字和已有的脏字数之和不超过 50 个，每个脏字的长度不能超过 200 字节。

应答

应答包体示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
10002	系统错误，请再次尝试或联系技术客服。
10003	请求命令非法，请再次尝试或联系技术客服。
10004	参数非法。请根据应答包中的 <code>ErrorInfo</code> 字段，检查必填字段是否填充，或者字段的填充是否满足协议要求。
10007	操作权限不足。请确认操作者是否是 App 管理员 。

接口调试工具

Web 调试工具

通过 [REST API 在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API 调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

REST API：查询 App 的自定义脏字（[v4/openim_dirty_words/get](#)）；

REST API：添加 App 的自定义脏字（[v4/openim_dirty_words/add](#)）。

全局禁言管理

设置全局禁言

最近更新时间：2018-04-19 18:22:54

功能说明

1. 设置账户的单聊消息全局禁言。如果账户被设置单聊消息全局禁言，在单聊消息禁言时间未到期时间内，所有的单聊消息发送失败，发送者接收到错误码 20012（详见 [错误码](#)），不会触发单聊消息之前回调(详见 [发单聊消息之前回调](#))。禁言时间到期后 IM 云通信后台自动解除单聊消息禁言，解除后所有单聊消息就能发送正常；对于永久全局单聊禁言，全局单聊消息禁言时间一直不过期。帐号默认未设置单聊消息全局禁言。
2. 设置账户的群组消息全局禁言。如果账户被设置群组消息全局禁言，在群组消息禁言时间内所有的群组消息发送失败，发送者接收到错误码 10017（详见 [错误码](#)），不会触发群内发言之前回调(详见 [群内发言之前回调](#))。禁言时间到期后 IM 通信后台自动解除群组消息禁言，解除后所有群组消息就能发送正常；对于永久全局群组消息禁言，全局群组消息禁言时间一直不过期。帐号默认未设置群组消息全局禁言。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openconfigsvr/setnospeaking?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式详见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP请求包体格式

JSON

请求包示例

```
{
  "Set_Account": "lumotuwe",
  "C2CmsgNospeakingTime": 4294967295, //C2CmsgNospeakingTime 和 GroupmsgNospeakingTime 是
  选填字段，但不能两个都不填
  "GroupmsgNospeakingTime": 7200
}
```

请求包字段说明

字段	类型	属性	说明
Set_Account	String	必填	设置禁言配置的帐号。
C2CmsgNospeakingTime	Number	选填	单聊消息禁言时间，秒为单位，非负整数，最大值为4294967295(十六进制 0xFFFFFFFF)。等于 0 代表取消账户禁言；等于最大值 4294967295(十六进制 0xFFFFFFFF)代表账户永久被设置禁言；其它代表该账户禁言时间。
GroupmsgNospeakingTime	Number	选填	群组消息禁言时间，秒为单位，非负整数，最大值为4294967295(十六进制0xFFFFFFFF)。等于 0 代表取消帐号禁言；最大值 4294967295(0xFFFFFFFF)代表账户永久禁言；其它代表该账户禁言时间。

应答

应答包体示例

```
{
  "ErrorCode": 0,
  "ErrorInfo": "",
}
```

应答包字段说明

字段	类型	说明
ErrorCode	Number	请求错误码。
ErrorInfo	String	错误相关信息。

错误码说明

除非发生网络错误(例如 502 错误),该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示。

公共错误码 (60000 到 79999) 详见 [公共错误码](#) 文档。

本 API 私有错误码如下：

错误码	含义说明
130001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
130004	Json 格式请求包中没有 Set_Account 字段。
130005	Json 格式请求包中没有 Set_Account 字段不是 String 类型。
130006	Json 格式请求包中 C2CmsgNospeakingTime 不是 String 类型。
130007	Json 格式请求包中 GroupmsgNospeakingTime 不是 String 类型。
130008	Json 格式请求包中 GroupmsgNospeakingTime 和 C2CmsgNospeakingTime 这两个字段都没有填写。

接口调试工具

Web 调试工具

通过 [REST API 在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具详见 [REST API调试工具](#)。

API 集成

无。

参考

查询全局禁言 ([v4/openconfigsvr/setnospeaking](#)) 。

查询全局禁言

最近更新时间：2018-04-19 18:23:10

背景说明

1. 全局禁言包括单聊消息禁言和全局消息禁言。
2. 如果账户被设置单聊消息全局禁言，在单聊消息禁言时间未到期时间内所有的单聊消息发送失败，到期后 IM 云通信后台自动解除单聊消息禁言，解除后所有单聊消息就能发送正常；对于永久全局单聊禁言，全局单聊消息禁言时间一直不过期。帐号默认未设置单聊消息全局禁言。
3. 如果账户被设置群组消息全局禁言，在群组消息禁言时间内所有的群组消息发送失败，到期后 IM 通信后台自动解除群组消息禁言，解除后所有群组消息就能发送正常；对于永久全局群组消息禁言，全局群组消息禁言时间一直不过期。帐号默认未设置群组消息全局禁言。

功能说明

1. 查询帐号的单聊消息全局禁言。
2. 查询帐号的群组消息全局禁言。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openconfigsvr/getnospeaking?usersig=xxx&identifier=admin&sdkappid=8888&random=9999&contenttype=json
```

请求参数

URL中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "Get_Account": "lumotuwe"
}
```

请求包字段说明

字段	类型	属性	说明
Get_Account	String	必填	查询禁言信息的帐号。

应答

应答包体示例

```
{
  "ErrorCode": 0,
  "ErrorInfo": "",
  "C2CmsgNospeakingTime": 4294967295,
  "GroupmsgNospeakingTime": 7196
}
```

应答包字段说明

注：如果被查询的用户不存在，通过该 REST API 获取的结果信息将会是永久禁言。

字段	类型	说明
ErrorCode	Number	请求错误码。
ErrorInfo	String	错误相关信息。
C2CmsgNospeakingTime	Number	单聊消息禁言时长，秒为单位，非负整数。等于 0 代表没有被设置禁言；等于最大值 4294967295(十六进制 0xFFFFFFFF)代表被永久设置禁言；其它代表该账户禁言时长，如果等于 3600 表示账户被禁言一小时。

字段	类型	说明
GroupmsgNospeakingTime	Number	群组消息禁言时长，秒为单位，非负整数。等于 0 代表没有被设置禁言；等于最大值 4294967295(十六进制 0xFFFFFFFF)代表被永久设置禁言；其它代表该账户禁言时长，如果等于 3600 表示账户被禁言一小时。

错误码说明

除非发生网络错误(例如 502 错误),该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码 (60000到79999) 参见 [公共错误码](#)。

本API私有错误码如下：

错误码	含义说明
130001	Json 格式解析失败,请检查请求包是否符合 JSON 规范。
130002	Json 格式请求包中没有 Get_Account 字段。
130003	Json 格式请求包中没有 Get_Account 字段不是 String 类型。

接口调试工具

Web 调试工具

通过 [REST API 在线调试工具](#) 调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API集成

无。

参考

设置全局禁言 ([v4/openconfigsvr/setnospeaking](#)) .

数据下载

消息记录下载

最近更新时间：2018-05-23 11:07:45

功能说明

App 管理员可以通过该接口获取 App 中某天某小时的所有单发或群组消息记录的下载地址。

注意事项

1. 消息记录以日志文件形式存在并 GZip 压缩，通过该接口获取到下载地址后，请自行下载并处理；
2. 消息记录文件每小时产生一次，例如 0 点（00:00~00:59）的数据在 01:00 后开始处理，一般 1 小时内处理完毕（消息较多则处理时间较长）；
3. 文件有效期 3 天，无论是否下载过，都会在 3 天后删除；
4. 获取到的下载地址存在有效期，请在过期前进行下载，若地址失效，请通过该接口重新获取；
5. 语音、文件和短视频消息，只能通过客户端接口拿到大数据资源。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/open_msg_svc/get_history?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据[工单模板](#)提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

```
{
  "ChatType": "C2C",
  "MsgTime": "2015120121"
}
```

请求包字段说明

字段	类型	属性	说明
ChatType	String	必填	消息类型，C2C:单发消息 Group:群组消息。
MsgTime	String	必填	需要下载的时间段，2015120121 表示获取 2015 年 12 月 1 日 21:00~21:59 的消息的下载地址。

应答

应答包体示例

```
{
  "File": [
    {
      "URL": "https://download.tim.qq.com/msg_history/2/9b8f8f063b73f61698ce11e58207e89ade40.gz",
      "ExpireTime": "2015-12-02 16:45:23",
      "FileSize": 65207,
      "FileMD5": "cceece008bb7f469a47cf8c4b7acb84e",
      "GzipSize": 1815,
      "GzipMD5": "c3a0269dde393fd7a8bb18bfdeae2e"
    }
  ],
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0
}
```

应答包字段说明

字段	类型	说明
----	----	----

字段	类型	说明
ActionStatus	String	请求处理的结果，“OK”表示处理成功，“FAIL”表示失败。
ErrorCode	Integer	错误码。
ErrorInfo	String	错误信息。
File	Array	消息记录文件下载信息。
URL	String	消息记录文件下载地址。
ExpireTime	String	下载地址过期时间，请在过期前进行下载，若地址失效，请通过该接口重新获取。
FileSize	Integer	GZip 压缩前的文件大小（单位 Byte）。
FileMD5	String	GZip 压缩前的文件 MD5。
GzipSize	Integer	GZip 压缩后的文件大小（单位 Byte）。
GzipMD5	String	GZip 压缩后的文件 MD5。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API 公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
1001	请求非法。请检查“请求URL”是否正确。
1002	参数非法。请检查是否管理员帐号，必填字段是否填充，或者字段的填充是否满足协议要求。
1003	系统错误。
1004	文件尚未生成，或者请求时段内没有消息。
1005	文件已过期。

消息记录文件格式

```
//单发消息
{"SdkAppId":1104620500,"ChatType":"C2C","MsgTime":"2015120121","MsgList":[
```

```
{"From_Account":"peakerdong","To_Account":"qiyueliuhuo2018","MsgTimestamp":1448974806,"MsgSeq":3452069198,"MsgRandom":45838,"MsgBody":[{"MsgType":"TIMTextElem","MsgContent":{"Text":"四等分"}}]},
{"From_Account":"group_root","To_Account":"group_test4","MsgTimestamp":1448974808,"MsgSeq":462709847,"MsgRandom":19196437,"MsgBody":[{"MsgType":"TIMTextElem","MsgContent":{"Text":"hi, beauty"}}]}
}

//群组消息
{"SdkAppId":1104620500,"ChatType":"Group","MsgTime":"2015120121","MsgList":[
{"From_Account":"Test_1","GroupId":"@TGS#1FDFVPAE2","MsgTimestamp":1448975384,"MsgSeq":1,"MsgBody":[{"MsgType":"TIMTextElem","MsgContent":{"Text":"Private activate"}}]},
{"From_Account":"Test_1","GroupId":"@TGS#1FDFVPAE2","MsgTimestamp":1448975384,"MsgSeq":1,"MsgBody":[{"MsgType":"TIMTextElem","MsgContent":{"Text":"Private activate"}}]}
]}
```

文件第一行记录了该文件的基本信息，接下来的每一行记录一条消息，直到最后一行"}"结束，每条消息格式可参考 [TIMMsgElement](#) 对象的定义；

若文件较小，可直接使用 JSON 库解析整个文件，MsgList 是该时段的消息数组。例如：

```
# Python示例代码
import gzip, json
with gzip.open('1104620500_Group_2015120121.gz', 'rb') as fp:
    info = json.load(fp)
    for msg in info['MsgList']:
        pass #do sth with msg
```

若文件较大，建议逐行解析。例如：

```
# Python示例代码
import gzip, json
with gzip.open('1104620500_Group_2015120121.gz', 'rb') as fp:
    cnt = -1
    for line in fp:
        line = line.strip().rstrip(b',')
        if line == b'}': break
        if cnt < 0:
            info = json.loads(line + b'}')
        else:
            msg = json.loads(line)
            #do sth with msg
        cnt += 1
```

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#) 调试本接口。

可能触发的回调

无

参考

无

Server SDK集成

PHP Server SDK

最近更新时间：2018-07-25 11:54:27

功能说明

1. PHP Server SDK 将一些常用的 REST API 封装成了函数，并以接口类的方式暴露给开发者，具体参见 `TimRestInterface.php` 文件中的 `TimRestInterface` 类；
2. 提供工具 `TimRestApiGear.php` 可直接访问 RestAPI。

PHP Server SDK 集成

[独立模式](#) 和 [托管模式](#) 的集成存在轻微差别，唯一的差别在于 `user_sig` 的设置方式。

1. 对于独立模式，需要调用 `generate_user_sig` 生成 `user_sig`；
2. 对于托管模式，需要先依照 [下载UserSig](#) 的指引下载 `user_sig`，然后再调用 `set_user_sig` 设置 `user_sig`。

API集成示例代码（独立模式）

```
// 设置 REST API 调用基本参数
$sdkappid = 1400000478;
$identifier = "myadmin";
$private_key_path = "/path/to/private/key";
$signature = "/signature/linux-signature64";

// 初始化API
$api = createRestAPI();
$api->init($sdkappid, $identifier);

// 生成签名，有效期一天
// 对于FastCGI，可以一直复用同一个签名，但是必须在签名过期之前重新生成签名
$ret = $api->generate_user_sig($identifier, '86400', $private_pem_path, $signature);
if ($ret == null)
{
    // 签名生成失败
    return -10;
}
// 调用api的成员函数，实现REST API的调用，以下单发消息，群组内发送文本消息，通用接口为示例：
// 单发消息
```

```
$ret = $api->openim_send_msg("myadmin", "lilei", "hello");

// 群组内发送消息
$ret = $api->group_send_group_msg("lilei", "@TGS#2QJXRPAE3", "hello");

// 通用接口, 在当前版本没有提供用户需要的REST API集成SDK的时候, 才考虑使用该接口,
// 用户可以使用该接口直接调用REST API, 需要用户构造REST API需要的请求包体
// 以调用REST API group_get_group_member_info 为例:
$req_body = array(
    "GroupId" => "@TGS#2QJXRPAE3", //群组ID
    "Limit" => 20, //Limit限制回包中群组的个数, 不得超过10000
    "Offset" => 5 //Offset控制从整个群组列表中的第多少个群组开始拉取信息
);
$ret = $api->comm_rest("group_open_http_svc", "group_get_group_member_info", $req_body);
```

API 集成示例代码（托管模式）

```
// 设置REST API调用基本参数
$sdkappid = 1400000478;
$identifier = "myadmin";
$user_sig = "eJx1zkFvgjAYxvE7...";

// 初始化API
$api = createRestAPI();
$api->init($sdkappid, $identifier);

// 设置签名
// 对于FastCGI, 可以一直复用同一个签名, 但是必须在签名过期之前重新下载签名
$ret = $api->set_user_sig($user_sig);
if ($ret == false)
{
    // 签名设置失败
    return -10;
}

// 调用api的成员函数, 实现REST API的调用, 以下单发消息, 群组内发送文本消息, 通用接口为示例:
// 单发消息
$ret = $api->openim_send_msg("myadmin", "lilei", "hello");

// 群组内发送消息
$ret = $api->group_send_group_msg("lilei", "@TGS#2QJXRPAE3", "hello");

// 通用接口, 在当前版本没有提供用户需要的REST API集成SDK的时候, 可以考虑使用该接口, 用户可以
// 通过该接口直接调用REST API的接口, 需要用户构造REST API要求的请求包体
// 以调用REST API group_get_group_member_info 为例:
```

```
$req_body = array(  
    "GroupId" => "@TGS#2QJXRPAE3", //群组ID  
    "Limit" => 20, //Limit限制回包中群组的个数, 不得超过10000  
    "Offset" => 5 //Offset控制从整个群组列表中的第多少个群组开始拉取信息  
);  
$ret = $api->comm_rest("group_open_http_svc", "group_get_group_member_info", $req_body);
```

TimRestApiGear.php 使用说明

独立模式

1. 配置 TimRestApiConfig.json 文件，其中：identifier 为 App 管理者账户；private_pem_path 为本地私钥位置；user_sig 请填写"。
2. 查看 signature 文件夹中对应脚本使用权限，如果无可执行权限，需要修改权限使其可被执行。
3. 执行 PHP TimRestApiGear.php 可看到该工具访问命令(用法)。详情请见：代码包中 README 文件。

托管模式

1. 配置 TimRestApiConfig.json 文件，其中：identifier 为App管理者账户；user_sig 为托管模式用户下载到的用户凭证，下载方式参见 [下载 UserSig](#)；private_pem_path 请填写"。
2. 执行 PHP TimRestApiGear.php 可看到该工具访问命令（用法）。详情请见：代码包中 README 文件。

SDK下载

您可通过如下两种方式下载：

1. [直接单击下载](#)；
2. 到 [github](#) 下载。

Node.js Server SDK

最近更新时间：2018-07-26 15:19:36

功能说明

1. Node.js Server SDK 将调用 REST API 的底层操作封装成 js 库，并以接口类的方式暴露给开发者，具体参见 lib 文件夹下 TimRestApi.js；
2. 提供 TimRestApiGear.js 示例工具，可直接访问 RestAPI；
3. 提供压力测试工具 TimSendGroupMsgPressTest.js，可测试在群内发送消息时，客户端接收消息能力。

Node.js Server SDK集成

支持 [独立模式](#)。

API 集成示例代码（独立模式）

该示例代码是 TimRestApiGear.js 的简化版本，详情可直接参考 TimRestApiGear.js。

```
var TimRestAPI = require('/path/lib/TimRestApi.js');
// 设置 REST API 调用基本参数
var config = {
  sdkAppid: '1400001111',
  identifier: 'admin',
  accountType: '106',
  version: '201512300000',
  privateKey: '/path/private.pem',
  expireAfter: 30 * 24 * 3600,
};

// 创建api对象
var api = new TimRestAPI(config);

// 调用api的成员函数，实现REST API的调用，以下单发消息，群组内发送文本消息为示例：
// 单发消息
api.init(function(err, data) {
  if (err) {
    // deal error
    console.log(err);
    return;
  }
}
```

```
var reqBody = {
  "To_Account": "86-18688732391",
  //消息接收者
  "From_Account": "86-18602833226",
  //消息发送者
  "MsgRandom": 123,
  //消息随机数
  "MsgBody": [{
    "MsgType": "TIMTextElem",
    //文本消息类型
    "MsgContent": {
      "Text": "hello" //具体文本消息
    }
  }]
}
api.request("openim", "sendmsg", reqBody,
function(err, data) {
  if (err) {
    console.log(err);
    return;
  }
  console.log(data);
});

// 群组内发送消息
api.init(function(err, data) {
  if (err) {
    // deal error
    console.log(err);
    return;
  }
  var reqBody = {
    GroupId: "UserDefinedGroupId",
    MsgBody: [{
      MsgType: "TIMTextElem",
      From_Account: "86-18602833226",
      MsgContent: {
        Text: "hello"
      }
    }]
  };
  api.request("group_open_http_svc", "send_group_msg", reqBody,
function(err, data) {
  if (err) {
```

```
console.log(err);  
return;  
}  
console.log(data);  
})  
});
```

群内发送消息压力测试脚本使用说明

在互动直播场景，消息量比较大时，由于弹幕等渲染操作，客户端可能会承受比较大的压力，建议在上线之前，用此工具来测试客户端性能。

使用方法：执行 `node TimSendGroupMsgPressTest.js` 可看到压测脚本工具访问命令(用法)。

注意事项：压测前需要确保群组存在，并且群内人数尽可能少，减少其他因素影响。

SDK 下载

您可通过如下两种方式下载：

1. [直接单击下载](#)；
2. 到 [github](#) 下载。

运营相关 监控配置

最近更新时间：2018-07-26 15:37:25

云监控概念

- [云监控概述](#)
- [云监控基本概念](#)

告警查询

入口在【云监控】-【我的告警】-【告警列表】。[链接>>](#)
告警列表展示发生过的基础告警信息。查看基础告警监控指标。

腾讯云 总览 云产品 点播 物理服务器 云数据库 弹性伸缩 运维管理VPN qcloud 帮助 工单 594

云监控 告警列表 当月短信配额已用 76 条，总数 1000 条 [告警策略使用指引](#)

新告警平台需配置告警规则，告警才会发送。旧告警数据会有部分缺失，属于正常情况。

基础告警 云拨测告警 自定义监控告警

2016-10-25 至 2016-10-26 请输入告警对象

发生时间 ↓	告警对象	告警内容	持续时长	告警状态	策略类型	策略名称	所属项目
找到 2 条结果。清除筛选条件							
2016-10-26 17:1...	10.104.116.219(...	CPU利用率 > 0 %	1小时29分钟	未恢复	云服务器策略	borishe_szjr_test	默认项目
2016-10-25 17:1...	10.104.116.219	CPU利用率 > 0 %	1天1小时29分钟	未恢复	云服务器策略	borishe_szjr_test	默认项目

告警策略

配置策略后，满足告警触发条件即发送告警至指定人。
只要 3 步，您就可以配置完成告警策略。以云通信为例说明。

新增告警策略

1) 进入“告警策略”页，单击新增告警按钮即可新增。告警策略列表中【已应用】字段展示该告警策略已关联的告警对象个数。

名称	触发条件	策略类型	已应用
test	带宽>0Mbps持续5分钟	CDN域名策略	0
默认	磁盘只读;ping不可达	默认云服务器策略	0
默认	磁盘只读;ping不可达	默认云服务器策略	0
默认	磁盘只读;ping不可达	默认云服务器策略	0

注：若已应用数值非 0，则不可删除该策略，解除所有告警对象与该策略关联后，才可删除该策略。

2) 选择告警触发条件。每个告警策略是一系列告警触发条件的集合。告警触发条件是“或”关系，即一个条件满足，就会发送告警。

策略名称：填写一个业务方可以识别的策略名称：如XX产品云通信监控策略

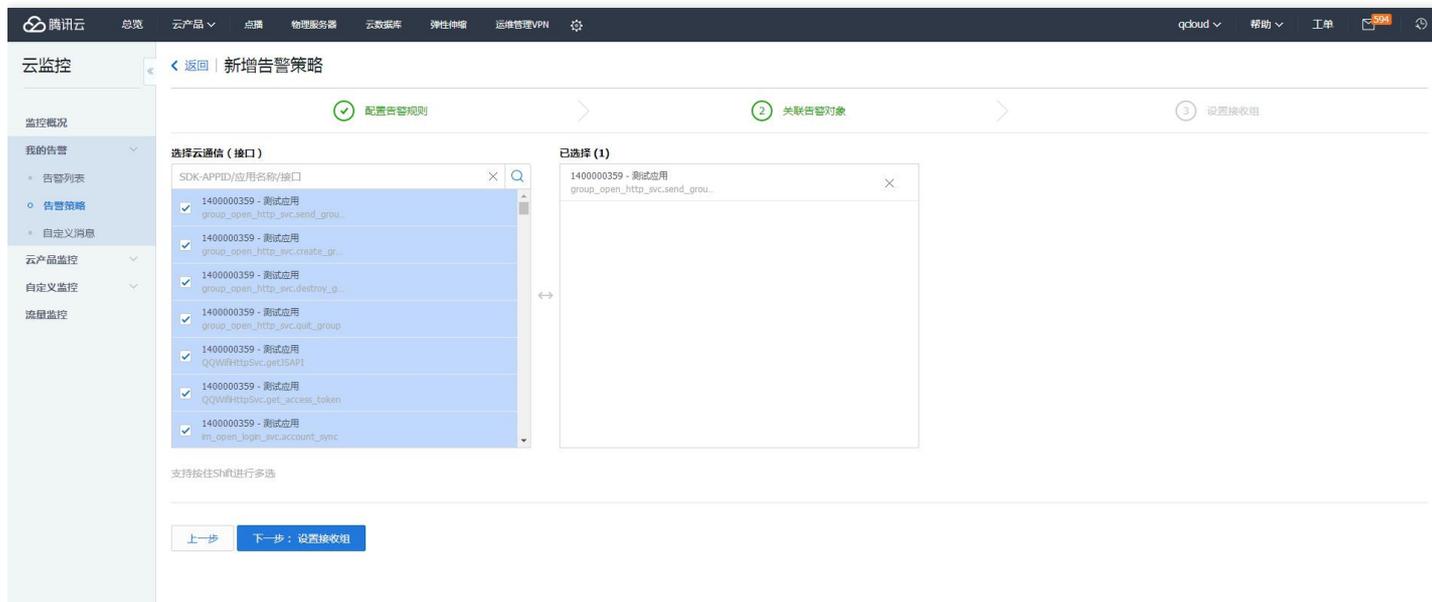
策略类型：选择云通信（接口）

统计周期：默认为 1 分钟

持续周期：选择多少个统计周期连续达到告警阈值才发送通知，建议选择一个周期

关联告警对象

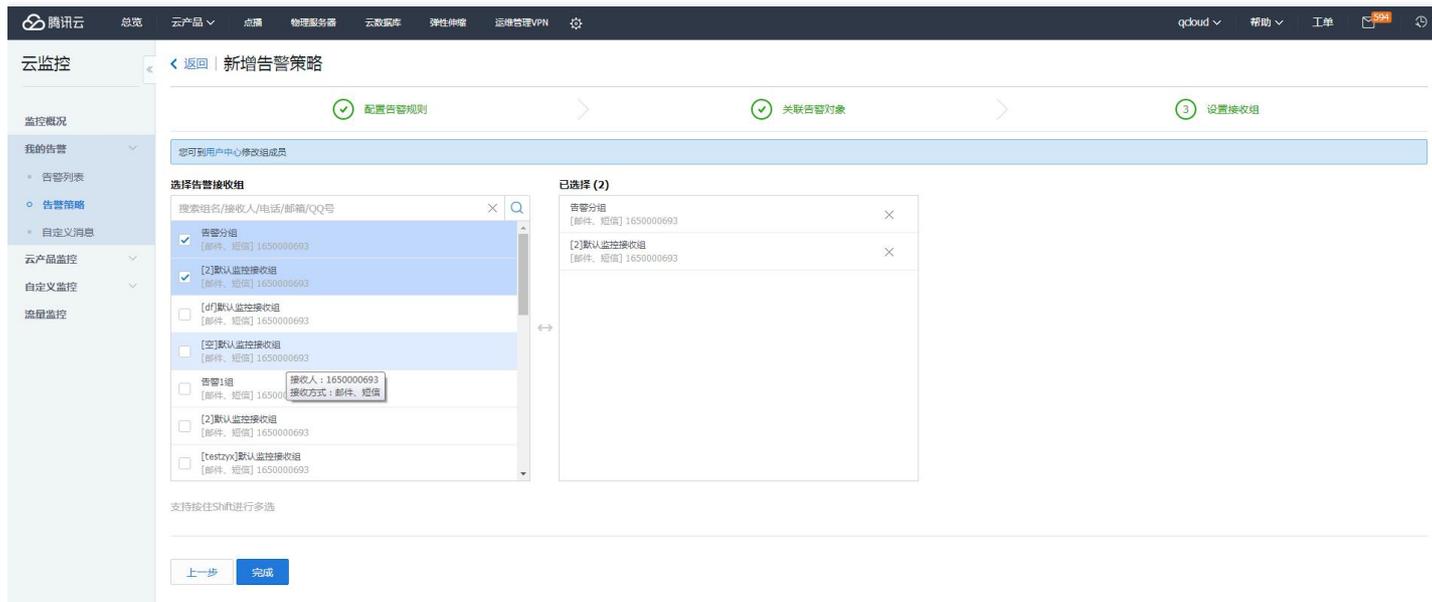
单击关联告警对象，进入管理告警策略子页面。



选择需要关联该策略的接口，支持按住 Shift 进行多选。

配置告警接收组

单击下一步，可以对告警接收组添加和删除。



完成这 3 步后，达到告警触发条件，您会马上收到告警。

告警接收组

您可以把关心相同告警的人聚合到一个组，配置一次，改变告警触发条件，组内的人员都会收到改变后的告警。

告警接收组		
管理告警接收组		
告警接收组	接收人	接收方式
test	2424242、bret、asf	邮件、短信

注：

- 告警策略中配置告警接收组，仅能添加或删除告警接收组与告警策略的关联，不能修改分组。
- 如果您需要修改组成员，或者告警接收渠道，您可以到【用户中心】-【权限设置】修改。如果您没有权限，可以联系具备全局协作者权限的人修改。

管理告警接收组 (tt)			×
选择告警接收组			您可到 用户中心 修改组信息
<input type="checkbox"/>	告警接收组	接收人	接收方式
<input type="checkbox"/>	test	无	邮件、短信
<input type="checkbox"/>	gAO	cdntest3	邮件、短信
<input type="checkbox"/>	告警分组	无	邮件、短信

告警接收方式

目前的告警接收方式支持邮件、短信。您可以到用户中心修改信息接收方式。

[用户中心指引>>](#)

取消告警订阅

若您不想收到某个策略的告警，即取消告警订阅。有 2 种办法：

- 1) 在对应策略里取消关联您所在的告警接收组。
- 2) 在【用户中心】-【权限设置】对应组里删除您的个人信息。

拉取运营数据

最近更新时间：2018-07-26 15:05:47

功能说明

App 管理员可以通过该接口拉取最近 30 天的运营数据，可拉取的字段见下文 [拉取运营数据](#)。

接口调用说明

请求 URL

```
https://console.tim.qq.com/v4/openconfigsvr/getappinfo?usersig=xxx&identifier=admin&sdkappid=88888888&random=99999999&contenttype=json
```

请求参数

URL 中各参数的含义以及填写方式参见 [REST API简介](#)。

最高调用频率

100次/秒。如需提升调用频率，请根据 [工单模板](#) 提交工单申请处理。

请求

HTTP 请求方式

POST

HTTP 请求包体格式

JSON

请求包示例

支持查询 appid 的最近 30 天的运营数据。

- **基础形式**

默认拉取所有字段

```
{}
```

- **指定拉取字段**

用 RequestField 指定需要拉取的字段。

```
{
  "RequestField":[
    "ChainIncrease",
    "ChainDecrease"
  ]
}
```

请求包字段说明

字段	类型	属性	说明
RequestField	Array	选填	该字段用来指定需要拉取的运营数据。不填默认拉取所有字段。详细可查看 可拉取的运营字段 。

应答

应答包体示例

- 基础形式

```
{
  "ErrorCode": 0,
  "ErrorInfo": "OK",
  "Result": [
    {
      "APNSMsgNum": "84",
      "ActiveUserNum": "2014",
      "AppId": "1104620500",
      "AppName": "实时通信场景体验版",
      "C2CAPNSMsgNum": "84",
      "C2CDownMsgNum": "11040",
      "C2CSendMsgUserNum": "9",
      "C2CUpMsgNum": "52209",
      "CallBackReq": "73069",
      "CallBackRsp": "72902",
      "ChainDecrease": "16",
      "ChainIncrease": "18",
      "Company": "林野",
      "Date": "20160607",
      "DownMsgNum": "11869",
      "GroupAPNSMsgNum": "0",
      "GroupAllGroupNum": "41913",
      "GroupDestroyGroupNum": "35019",
      "GroupDownMsgNum": "829",
```

```

"GroupJoinGroupTimes": "121438",
"GroupNewGroupNum": "35904",
"GroupQuitGroupTimes": "108292",
"GroupSendMsgGroupNum": "5189",
"GroupSendMsgUserNum": "12",
"GroupUpMsgNum": "8433",
>LoginTimes": "13708",
>LoginUserNum": "2094",
"MaxOnlineNum": "62",
"QQ": "2819697038",
"RegistUserNumOneDay": "1052",
"RegistUserNumTotal": "53091",
"SendMsgUserNum": "19",
"TextMsgInNum": "2",
"TextMsgOutNum": "0",
"UpMsgNum": "60642",
"VoiceMsgInNum": "0",
"VoiceMsgOutNum": "0"
}
    
```

• 指定拉取字段

```

{
  "ErrorCode":0,
  "ErrorInfo":"OK",
  "Result":[{"
    "ChainDecrease":"8",
    "ChainIncrease":"8",
    "Date":"20160605"
  },
  {
    "ChainDecrease":"17",
    "ChainIncrease":"17",
    "Date":"20160604"
  }
]
}
    
```

应答包字段说明

字段	类型	说明
Result	Array	请求最近30天的运营数据。
ErrorCode	Integer	错误码。

字段	类型	说明
ErrorInfo	String	错误信息。

错误码说明

除非发生网络错误（例如 502 错误），该接口的 HTTP 返回码均为 200。真正的错误码、错误信息是通过应答包体中的 ErrorCode、ErrorInfo 来表示的。

公共错误码（60000 到 79999）参见 [REST API公共错误码](#)。

本 API 私有错误码如下：

错误码	含义说明
130001	请求包 json 解析错误。
130009	sql open 错误。
130010	sql ping 错误。
130011	sql 查询错误。
130012	sql 结果解析错误。

可拉取的运营字段

字段	说明
AppName	应用名称
AppId	应用 AppId
Company	所属客户名称
QQ	所属客户 QQ
ActiveUserNum	活跃用户数
RegistUserNumOneDay	新增注册人数
RegistUserNumTotal	累计注册人数
LoginTimes	登录次数
LoginUserNum	登录人数
UpMsgNum	上行消息数

字段	说明
DownMsgNum	下行消息数
SendMsgUserNum	发消息人数
APNSMsgNum	APNS 推送数
C2CUpMsgNum	上行消息数 (C2C)
C2CDownMsgNum	下行消息数 (C2C)
C2CSendMsgUserNum	发消息人数 (C2C)
C2CAPNSMsgNum	APNS 推送数 (C2C)
MaxOnlineNum	最高在线人数
ChainIncrease	关系链对数增加量
ChainDecrease	关系链对数删除量
GroupUpMsgNum	上行消息数 (群)
GroupDownMsgNum	下行消息数 (群)
GroupSendMsgUserNum	发消息人数 (群)
GroupAPNSMsgNum	APNS推送数 (群)
GroupSendMsgGroupNum	发消息群组数
GroupJoinGroupTimes	入群总数
GroupQuitGroupTimes	退群总数
GroupNewGroupNum	新增群组数
GroupAllGroupNum	累计群组数
GroupDestroyGroupNum	解散群个数
TextMsgInNum	文字短信国内下发量
TextMsgOutNum	文字短信国外下发量
VoiceMsgInNum	国内语音验证码下发回调用量
VoiceMsgOutNum	国外语音验证码下发回调用量

字段	说明
CallBackReq	回调请求数
CallBackRsp	回调应答数

接口调试工具

Web 调试工具

通过 [REST API在线调试工具](#)调试本接口。

Server 调试工具

无。

更多调试工具参见 [REST API调试工具](#)。

API 集成

PHP 集成

无。

可能触发的回调

无。

参考

无。