

# 云通信

## 第三方回调

### 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

### 第三方回调

第三方回调简介

第三方回调接入指引

回调命令列表

单发消息相关回调

发单聊消息之前回调

发单聊消息之后回调

群组系统相关回调

群内发言之前回调

群内发言之后回调

创建群组之前回调

创建群组之后回调

申请入群之前回调

拉人入群之前回调

新成员入群之后回调

群成员离开之后回调

群组满员之后回调

群组解散之后回调

群组资料修改之后回调

在线状态相关回调

状态变更回调

关系链系统相关回调

添加好友之前回调

添加好友之后回调

添加好友回应之前回调

删除好友之后回调

添加黑名单之后回调

删除黑名单之后回调

回调双向认证配置指南

Apache双向认证配置指南

Nginx双向认证配置指南

# 第三方回调

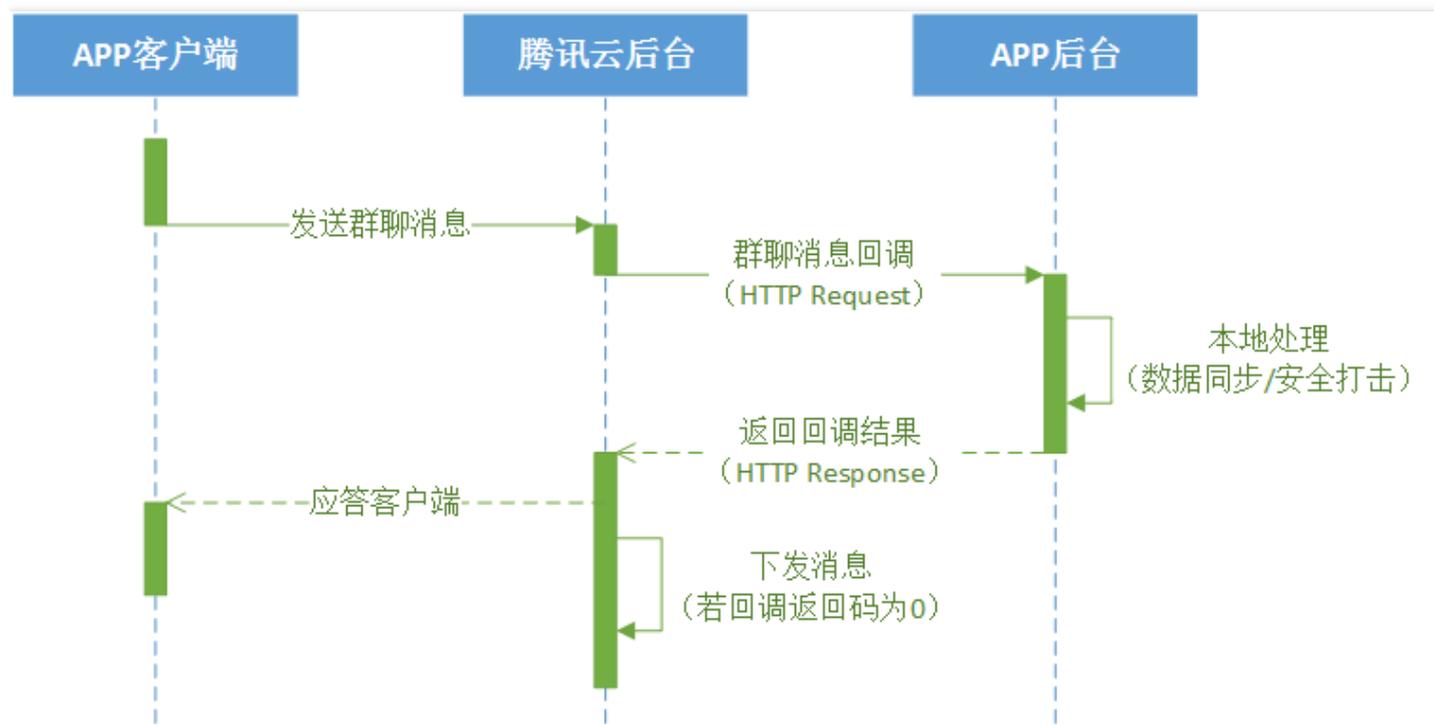
## 第三方回调简介

最近更新时间：2018-04-03 14:59:18

### 1 第三方回调简介

为方便第三方APP精细化控制APP的功能形态，腾讯云为第三方提供了完全免费且强大的回调能力。所谓回调，即腾讯云会在某一事件发生之前或者之后，向APP的后台服务器发送请求，APP后台可以据此进行必要的数据库同步，或者干预事件的后续处理流程。

例如，APP可以开启群发消息回调，腾讯云后台会在下发该消息之前回调第三方服务器，并根据回调结果决定该消息是否应当下发。APP不仅可以基于该回调来实现消息同步，而且可以进行发消息安全打击。其业务流程如下图所示：



### 2 回调分类

从功能角度来看，回调可以分为如下四大类：

1. 单发消息回调；
2. 群组相关回调；

3. 用户资料相关回调；
4. 关系链相关回调；
5. 状态变更回调。

从腾讯云对回调结果的处理角度来看，回调可以分为两大类：

1. 事件发生之前回调：回调的主要目的在于让第三方APP后台可以干预该事件的处理逻辑，腾讯云会根据回调返回码确定后续处理流程（例如发送群消息之前回调）；
2. 事件发生之后通知：回调的主要目的在于让第三方APP实现必要的的数据同步，腾讯云忽略回调返回码（例如群组成员退群之后通知）。

## 3 回调协议

第三方回调基于HTTP/HTTPS协议，APP服务器需要向腾讯云提供回调URL，腾讯云使用POST请求的方式来向APP服务器发起回调请求。

腾讯云在发起回调时，会在APP提供的URL之后增加如下几个参数：

参数名称	参数含义
SdkAppid	APP在云通信中分配到的ID。
CallbackCommand	回调命令字。
contenttype	可选，通常为json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。对应不同的平台类型，可能的取值有： RESTAPI（使用REST API发送请求）、Web（使用Web SDK发送请求）、 Android、iOS、Windows、Mac、Unkown（使用未知类型的设备发送请求）。

具体的回调内容则会包含在HTTP请求包体中，参见回调示例。

### 3.1 回调示例

回调请求示例：

```
POST /?SdkAppid=888888&CallbackCommand=Group.CallbackAfterNewMemberJoin&contenttype=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform HTTP/1.1
```

```
Host: www.example.com
```

```
Content-Length: 337
```

```
{
```

```
"CallbackCommand": "Group.CallbackAfterNewMemberJoin",
"GroupId": "@TGS#2J4SZEAE",
"Type": "Public",
"JoinType": "Apply",
"Operator_Account": "leckie",
"NewMemberList": [
{
"Member_Account": "jared"
},
{
"Member_Account": "tommy"
}
]
```

回调应答示例：

```
HTTP/1.1 200 OK
Server: nginx/1.7.10
Date: Fri, 09 Oct 2015 02:59:55 GMT
Content-Length: 75

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

## 4 安全考虑

腾讯云支持三种回调类型：

1. HTTP回调；
2. HTTPS回调，APP服务器的webserver配置的是CA机构签发的证书或者是腾讯云免费签发的证书；
3. HTTPS双向认证回调，APP服务器的webserver配置的是CA机构签发的证书或者是腾讯云免费签发的证书，且启用双向认证能力。

三种方案的安全性逐步递增：

1. HTTP回调存在两个缺陷：一是明文传输的数据容易被窃听，二是第三方APP无法判断回调请求是否真正来自于腾讯云；
2. 对于HTTPS回调，如果不启用双向认证，可以解决数据的加密问题，但依然无法确保回调的请求来源是腾讯云；

3. 只有HTTPS与双向认证结合，才能确保第三方回调的安全性。

因此我们强烈建议APP使用第三种方式实现回调，且腾讯云签发证书完全免费。

## 5 回调超时时间

腾讯云回调APP后台服务器的超时时间为2秒，且没有重试。如果回调超时，后续处理逻辑与没有配置回调时相同（例如，假设“发送群消息之前回调”超时，消息会正常下发）。

为确保回调成功率，第三方APP应当尽可能加快回调处理速度，例如先发送回调应答，然后再处理具体业务逻辑。

## 6 回调配置

目前云通信控制台支持自助配置回调，包括配置回调URL以及启用哪些回调。配置方法参见[第三方回调接入指引](#)。

### 注意：

在控制台中自助配置的回调仅支持HTTP/HTTPS回调。如果您需要启用安全级别最强的HTTPS双向认证，需要：

1. 在控制台中配置回调URL（必须为HTTPS域名）、回调开关；
2. 给腾讯云提需求工单，由云通信后台给APP签发双向认证所需的证书，所需要的资料包括：
  - i. sdkappid；
  - ii. APP名称；
  - iii. 回调URL（必须与控制台中自助配置的回调URL一致）；
3. 拿到证书之后，依照如下两篇指引配置HTTPS双向认证：
  - i. [Apache配置HTTPS双向认证指南](#)；
  - ii. [Nginx配置HTTPS双向认证指南](#)。

# 第三方回调接入指引

最近更新时间：2018-07-26 11:53:29

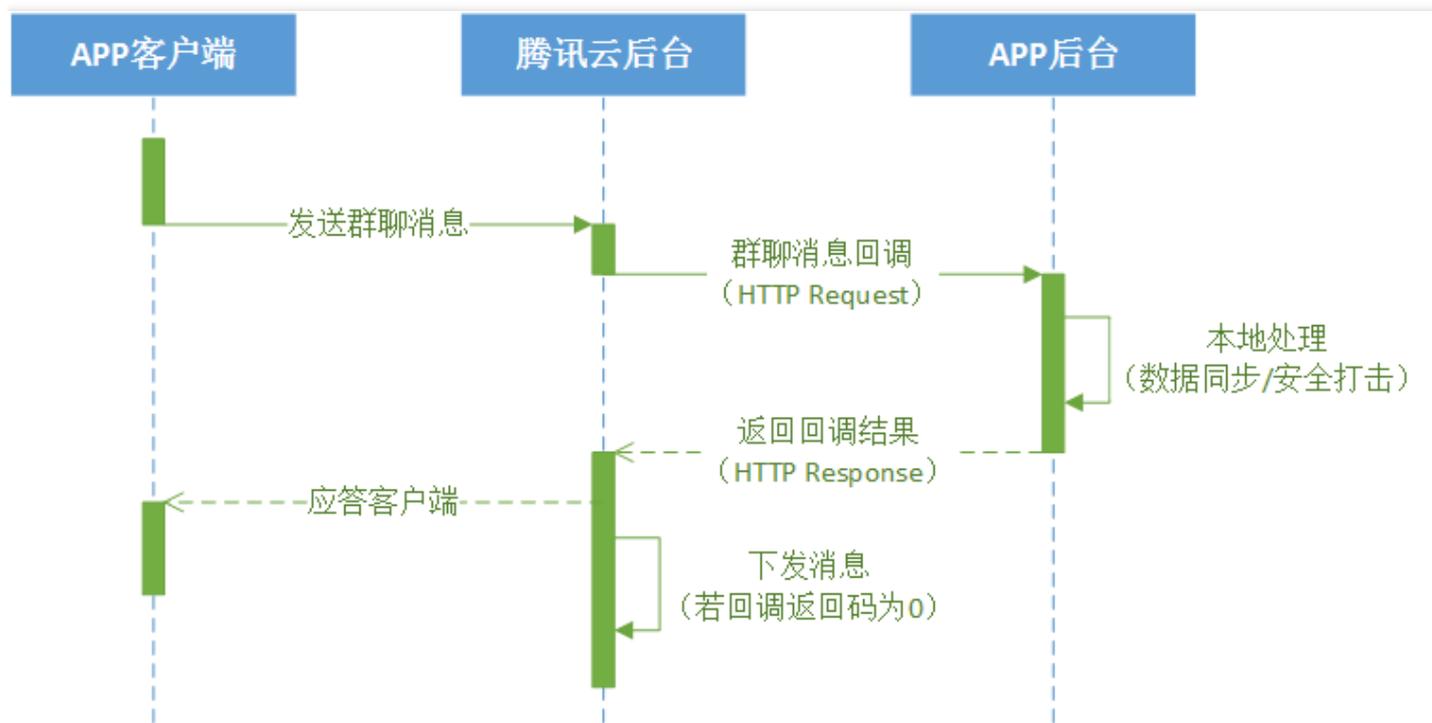
## 内容提要

所谓 **第三方回调**，即云通信后台会在某一事件发生之前或者之后，向 App 的后台服务器发送请求，App 后台可以据此进行必要的数据库同步，或者干预事件的后续处理流程。云通信目前支持的回调请参见 [回调命令列表](#)。在接下来的内容中，我们使用如下案例来介绍接入第三方回调的流程：

1. 回调 URL 为 `https://www.example.com/imcallback`。
2. 需要启用【群内发言之前回调】以及【单发消息之前回调】，目的是实时记录 App 中用户的发消息情况。

## 回调方式

第三方回调将通过 HTTP/HTTPS 请求的方式发送给 App 后台服务器，回调链路的安全问题参见 [安全考虑](#)。App 后台服务器需要处理云通信的回调请求并尽快进行应答。以 [群内发言之前回调](#) 为例，回调业务流程如下图所示。



## 如何配置回调

回调的配置包含两个方面，分别为：配置回调 URL、选择启用哪些回调。这两方面的配置均可以在控制台中自助完成。在控制台中单击应用最右侧的【更多】-【回调配置】：



进入如下页面：

基础配置

SPEAR引擎配置

用户管理

群组管理

回调配置

## 回调URL

添加回调URL,并进行事件回调配置。

## 事件回调配置 [编辑](#)

### 群组

- 创建群之前     创建群之后     申请加群之前     拉人入群之后     新成员入群之后
- 修改群名称之后     修改群介绍之后     修改群公告之后     修改群头像URL之后

### 资料关系链

- 添加好友     删除好友     添加黑名单     删除黑名单

### C2C

- C2C消息回调

## 配置回调 URL

单击【添加回调 URL】来配置回调 URL。完成回调 URL 的配置之后，您还可以单击【更改URL】来变更回调 URL。

### 注意：

- 回调 URL 必须以 `http://` 或者 `https://` 开头。
- 如果您暂未申请域名，亦可直接配置 IP，例如：`http://123.123.123.123/imcallback`。
- 回调链路的安全强度有三种：① HTTP 回调，② HTTPS 回调，③ HTTPS 双向认证回调，安全强度依次增强（详见 [安全考虑](#)）；目前暂不支持 HTTPS 双向认证回调的自助配置。如果您需要启用 HTTPS 双向认证以达到最高的安全级别，请联系商务经理提 [需求工单](#)。

## 选择启用哪些回调

单击【事件回调配置】后的【编辑】按钮，选中【群内发消息之前】和【C2C 消息回调】，并保存。配置完成后，回调的状态如下所示，至此，第三方回调便配置完成，将在 5 分钟内生效。



## 回调不通时如何自助排查问题

如果遇到回调不通的问题，建议 App 先依照如下清单排查一下自己的回调服务是否存在问题。

回调不通的症状	可能存在的原因
回调 URL 访问超时	1.云通信后台无法完成 DNS 解析，请确认该域名是否在公网生效。（例如，回调 HOST 为 <code>http://notexist.com</code> ，该域名不存在，通信后台无法完成 DNS 解析。） 2.云通信无法访问到回调 URL 中配置的 IP，请确认该 IP 是否从公网可达。（例如，回调 HOST 为 <code>http://10.0.0.1</code> ，该域名为 App 内网 IP，云通信后台无法访问到该 IP。） 3.App 回调服务防火墙策略限制，请检查防火墙配置。（例如，App 回调服务器拒绝了所有到达 80 端口的请求。）
回调服务拒绝访问	云通信后台可以访问到 HOST，但链接建立失败，请确认 WebServer 已经正确启动。（例如：App 回调服务器的 WebServer 并未启动，或者端口配置错误。）
回调服务 HTTPS 证书配置错误	回调方式为 HTTPS（或 HTTPS 双向认证），云通信后台能够访问到 App 回调服务器，但判定 App WebServer 配置的证书非法。请确定 HTTPS 证书配置正确。
回调服务 HTTPS 双向认证配置错误	回调方式为 HTTPS 双向认证，云通信校验 App 回调服务器的证书合法，但 App 回调服务器校验云通信的证书失败。
回调服务 HTTP 返回码非 200	回调请求成功，但应答报文中的 HTTP 返回码非 200。

回调不通的症状	可能存在的原因
回调应答包体解析失败	回调请求包体非 JSON 格式。

# 回调命令列表

最近更新时间：2018-08-14 15:41:01

## 单发消息

回调类型	回调命令
发消息之前回调	<a href="#">C2C.CallbackBeforeSendMsg</a>
发消息之后回调	<a href="#">C2C.CallbackAfterSendMsg</a>

## 群组系统

回调类型	回调命令
创建群组之前回调	<a href="#">Group.CallbackBeforeCreateGroup</a>
创建群组之后回调	<a href="#">Group.CallbackAfterCreateGroup</a>
申请入群之前回调	<a href="#">Group.CallbackBeforeApplyJoinGroup</a>
拉人入群之后回调	<a href="#">Group.CallbackBeforeInviteJoinGroup</a>
新成员入群之后回调	<a href="#">Group.CallbackAfterNewMemberJoin</a>
群成员离开之后回调	<a href="#">Group.CallbackAfterMemberExit</a>
群内发言之前回调	<a href="#">Group.CallbackBeforeSendMsg</a>
群内发言之后回调	<a href="#">Group.CallbackAfterSendMsg</a>
群组满员之后回调	<a href="#">Group.CallbackAfterGroupFull</a>
群组解散之后回调	<a href="#">Group.CallbackAfterGroupDestroyed</a>
群组资料修改之后回调	<a href="#">Group.CallbackAfterGroupInfoChanged</a>

## 关系链系统

回调类型	回调命令
添加好友之后回调	<a href="#">Sns.CallbackFriendAdd</a>
删除好友之后回调	<a href="#">Sns.CallbackFriendDelete</a>
添加黑名单之后回调	<a href="#">Sns.CallbackBlackListAdd</a>
删除黑名单之后回调	<a href="#">Sns.CallbackBlackListDelete</a>

## 在线状态

回调类型	回调命令
状态变更回调	<a href="#">State.StateChange</a>

# 单发消息相关回调

## 发单聊消息之前回调

最近更新时间：2017-11-18 14:38:32

## 1 功能说明

APP后台可以通过该回调实时监控用户的c2c消息，包括：

1. 对c2c消息进行实时记录（例如记录日志，或者同步到其他系统）；
2. 拦截用户的c2c发言请求；
3. 修改用户发言内容（例如敏感词过滤，或者增加一些APP自定义信息）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 若同时开启发单聊消息之前和之后两种回调，且发单聊消息之前回调返回禁止发言，则发单聊消息之后回调将不会被触发；
5. 若同时开启发单聊消息之前和之后两种回调，且发单聊消息之前回调修改了消息体，则发单聊消息之后回调将使用修改过的消息进行回调；
6. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发送c2c消息；
2. APP管理员通过REST API发送c2c消息。

## 4 回调发生时机

云通信后台收到用户发送的c2c消息之后、将该消息下发给目标用户之前。

## 5 接口说明

## 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com` 。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：C2C.CallbackBeforeSendMsg。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包体示例

注意，回调请求包体位于HTTP POST请求的Body中

```
{
  "CallbackCommand": "C2C.CallbackBeforeSendMsg", // 回调命令
  "From_Account": "jared", // 发送者
  "To_Account": "Jonh", // 接收者
  "MsgBody": [ // 消息体，参见TIMMessage消息对象
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    }
  ]
}
```

## 5.6 回调请求包体字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
From_Account	String	消息发送者ID。
To_Account	String	消息接收者ID。
MsgBody	Array	消息体，具体参见 <a href="#">消息格式描述</a> 。注意，当前仅支持文本、表情、文件、位置和自定义消息体回调。

## 5.7 回调应答包示例（允许发言）

允许用户发言，同时也不修改即将下发的消息的内容。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 0为允许发言
}
```

## 5.8 回调应答包示例（禁止发言）

不允许用户发言，该消息将不会下发，同时给调用方返回错误码：20006。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 1 // 1为拒绝发言
}
```

## 5.9 回调应答包示例（修改消息内容）

如下的应答示例为：对用户发送的c2c消息进行了修改（增加了自定义消息）。云通信后台将会下发经过修改之后的消息。

APP后台可以基于这一特性在用户发送的消息中增加一些特殊内容，比如用户等级、头衔等信息。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0, // 必须为0，只有这样，修改之后的消息才能正常下发
  "MsgBody": [ // APP修改之后的消息，如果没有，则默认使用用户发送的消息

```

```

{
  "MsgType": "TIMTextElem", // 文本
  "MsgContent": {
    "Text": "red packet"
  }
},
{
  "MsgType": "TIMCustomElem", // 自定义消息
  "MsgContent": {
    "Desc": " CustomElement.MemberLevel ", // 描述
    "Data": " LV1" // 数据
  }
}
]
}
    
```

## 5.10 回调应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0为允许发言；1为拒绝发言。若业务希望拒绝发言的同时，将错误码ErrorCode和ErrorInfo传递至客户端，请将错误码ErrorCode设置在[120001, 130000]区间内。
ErrorInfo	String	必填	错误信息。
MsgBody	Array	选填	经过APP修改之后的消息体，云通讯后台将把修改后的消息发送给好友，具体格式参见 <a href="#">消息格式描述</a> 。

## 6 参考

[第三方回调简介](#)；

REST API：[单发单聊消息](#)，[批量发单聊消息](#)；

[发单聊消息之后回调](#)。

# 发单聊消息之后回调

最近更新时间：2017-11-18 14:38:46

## 1 功能说明

APP后台可以通过该回调实时监控用户的c2c消息，包括：

1. 对c2c消息进行实时记录（例如记录日志，或者同步到其他系统）；
2. 对c2c消息进行数据统计（例如人数，消息数等）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 若同时开启发单聊消息之前和之后两种回调，且发单聊消息之前回调返回禁止发言，则发单聊消息之后回调将不会被触发；
5. 若同时开启发单聊消息之前和之后两种回调，且发单聊消息之前回调修改了消息体，则发单聊消息之后回调将使用修改过的消息进行回调；
6. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发送c2c消息；
2. APP管理员通过REST API发送c2c消息。

## 4 回调发生时机

云通信后台收到用户发送的c2c消息并将该消息下发给目标用户之后。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com` 。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：C2C.CallbackAfterSendMsg。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包体示例

注意，回调请求包体位于HTTP POST请求的Body中

```
{
  "CallbackCommand": "C2C.CallbackAfterSendMsg", // 回调命令
  "From_Account": "jared", // 发送者
  "To_Account": "Jonh", // 接收者
  "MsgBody": [ // 消息体，参见TIMMessage消息对象
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    }
  ]
}
```

## 5.6 回调请求包体字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
From_Account	String	消息发送者ID。
To_Account	String	消息接收者ID。
MsgBody	Array	消息体，具体参见 <a href="#">消息格式描述</a> 。

## 5.7 回调应答包示例

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 0为回调成功，1为回调出错
}
```

## 5.8 回调应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0为回调成功；1为回调出错。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)；

REST API：[单发单聊消息](#)，[批量发单聊消息](#)；

[发单聊消息之前回调](#)。

# 群组系统相关回调

## 群内发言之前回调

最近更新时间：2017-03-09 10:56:55

## 1 功能说明

APP后台可以通过该回调实时监控用户的群发消息，包括：

1. 对群消息进行实时记录（例如记录日志，或者同步到其他系统）；
2. 拦截用户在群内发言的请求；
3. 修改用户发言内容（例如敏感词过滤，或者增加一些APP自定义信息）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发送群消息；
2. APP管理员通过REST API发送群组消息。

## 4 回调发生时机

云通信后台收到用户发送的群发消息之后、将该消息下发给群成员之前。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackBeforeSendMsg。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackBeforeSendMsg", // 回调命令
  "GroupId": "@TGS#2J4SZEAE", // 群组ID
  "Type": "Public", // 群组类型
  "From_Account": "jared", // 发送者
  "Operator_Account": "admin", // 请求的发起者
  "Random": 123456, // 随机数
  "MsgBody": [ // 消息体，参见TIMMessage消息对象
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
GroupId	String	产生群消息的群组ID。
Type	String	产生群消息的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
From_Account	String	消息发送者ID。
Operator_Account	String	请求发起者ID，可以用来识别是否为管理员请求的。
Random	Integer	发消息请求中的32位随机数。
MsgBody	Array	消息体，具体参见 <a href="#">消息格式描述</a> 。

## 5.7 回调应答包示例

### 5.7.1 允许发言

允许用户发言，同时也不修改即将下发的消息的内容。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 0为允许发言
}
```

### 5.7.2 禁止发言

不允许用户发言，该消息将不会下发，同时给调用方返回错误码：10016。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 1 // 1为拒绝发言
}
```

### 5.7.3 静默丢弃

不允许用户发言，该消息将不会下发，但会给调用方返回成功，使调用方以为消息已经发出。

```
{
  "ActionStatus": "OK",
```

```
"ErrorInfo": "",
"ErrorCode": 2 // 2为静默丢弃
}
```

#### 5.7.4 修改消息内容

如下的应答示例为：对用户发送的群消息进行了修改（增加了自定义消息）。云通信后台将会下发经过修改之后的消息。

APP后台可以基于这一特性在用户发送的消息中增加一些特殊内容，比如用户等级、头衔等信息。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0, // 必须为0, 只有这样, 修改之后的消息才能正常下发
  "MsgBody": [ // APP修改之后的消息, 如果没有, 则默认使用用户发送的消息
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    },
    {
      "MsgType": "TIMCustomElem", // 自定义消息
      "MsgContent": {
        "Desc": "CustomElement.MemberLevel", // 描述
        "Data": "LV1" // 数据
      }
    }
  ]
}
```

### 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0为允许发言；1为拒绝发言；2为静默丢弃。
ErrorInfo	String	必填	错误信息。

MsgBody	Array	选 填	经过APP修改之后的消息体，云通讯后台将把修改后的消息发送到群组中，具体格式参见 <a href="#">消息格式描述</a> 。
---------	-------	--------	---

## 6 参考

[第三方回调简介](#)；

REST API：[在群组中发送普通消息](#)。

# 群内发言之后回调

最近更新时间：2018-05-31 15:56:46

## 功能说明

App 后台可以通过该回调实时监控用户的群发消息，包括：通知 App 后台有群组消息发送成功，App 可以据此进行必要的数据库同步。

## 注意事项

1. 要启用回调，必须配置回调 URL，并打开本条回调协议对应的开关，配置方法详见 [第三方回调配置](#)。
2. 回调的方向是：云通讯后台向 App 后台发起 HTTP POST 请求。
3. App 后台在收到回调请求之后，务必校验请求 URL 中的参数 SdkAppid 是否是自己的 SdkAppid。
4. 其他安全相关事宜请参考 [第三方回调：安全考虑](#)。

## 可能触发该回调的场景

1. App 用户通过客户端发送群消息。
2. App 管理员通过 REST API 发送群组消息。

## 回调发生时机

群组消息发送成功之后。

## 接口说明

### 请求 URL

示例中 App 配置的回调 URL 为 `https://www.example.com`。

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatfor
```

### 请求参数

参数	含义
SdkAppid	App 在云通讯申请的 Appid
CallbackCommand	固定为：Group.CallbackAfterSendMsg
contenttype	固定为：json
ClientIP	客户端 IP 地址
OptPlatform	客户端平台，取值参见 <a href="#">回调协议</a> 中 OptPlatform 的参数含义

## HTTP 请求方式

HTTP 请求方式：POST。

## HTTP 请求包体格式

HTTP 请求包体格式：JSON。

## 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterSendMsg", // 回调命令
  "GroupId": "@TGS#2J4SZEAE", // 群组 ID
  "Type": "Public", // 群组类型
  "From_Account": "jared", // 发送者
  "MsgSeq": 123, // 消息的序列号
  "MsgTime": 1490686222, // 消息的时间
  "MsgBody": [ // 消息体，参见 TIMMessage 消息对象
    {
      "MsgType": "TIMTextElem", // 文本
      "MsgContent": {
        "Text": "red packet"
      }
    }
  ]
}
```

## 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令

字段	类型	说明
GroupId	String	产生群消息的群组 ID
Type	String	产生群消息的 <a href="#">群组类型</a> ，例如 Private，Public 和 ChatRoom
From_Account	String	消息发送者 ID
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
MsgBody	Array	消息体，具体参见 <a href="#">消息格式描述</a>

## 回调应答包示例

App 后台同步数据后，发送回调应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 //忽略回调结果
}
```

## 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK 表示处理成功，FAIL 表示失败
ErrorCode	Integer	必填	错误码，此处填 0 表示忽略应答结果
ErrorInfo	String	必填	错误信息

## 参考

- [第三方回调简介](#)
- REST API：[在群组中发送普通消息](#)

# 创建群组之前回调

最近更新时间：2017-03-09 09:52:23

## 1 功能说明

APP后台可以通过该回调实时监控用户的创建群组消息，包括：

1. 后台可以拒绝用户创建群组的请求。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端创建群组；
2. APP管理员通过REST API创建群组。

## 4 回调发生时机

客户端或者APP管理员通过REST API发起创建群组请求之后、云通信后台准备创建群组之前。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackBeforeCreateGroup。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackBeforeCreateGroup", // 回调命令
  "Operator_Account": "leckie", // 操作者
  "Owner_Account": "leckie", // 群主
  "Type": "Public", // 群组类型
  "Name": "MyFirstGroup", // 群组名称
  "CreatedNum": 123, // 该用户已创建的同类的群组个数
  "MemberList": [ // 初始成员列表
    {
      "Member_Account": "bob"
    },
    {
      "Member_Account": "peter"
    }
  ]
}
```

### 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
Operator_Account	String	发起创建群组请求的操作者ID。

Owner_Account	String	请求创建的群的群主ID。
Type	String	产生群消息的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Name	String	请求创建的群组的名称。
CreatedNum	Integer	该用户已创建的同类的群组个数。
MemberList	Array	请求创建的群组的初始化成员列表。

## 5.7 回调应答包示例

### 5.7.1 允许创建

允许用户创建群组。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 允许创建
}
```

### 5.7.2 禁止创建

不允许用户创建群组，群组将不会被创建，同时给调用方返回错误码：10016。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 1 // 拒绝创建
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0为允许创建；1为拒绝创建。
ErrorInfo	String	必填	错误信息。

---

## 6 参考

[第三方回调简介](#)；

REST API：[创建群组](#)。

# 创建群组之后回调

最近更新时间：2017-03-09 02:13:11

## 1 功能说明

APP后台可以通过该回调实时监控用户的创建群组消息，包括：

1. 通知APP后台有群组创建成功，APP后台可以据此进行数据同步等操作。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端创建群组成功；
2. APP管理员通过REST API创建群组成功。

## 4 回调发生时机

群组创建成功之后。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterCreateGroup。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```

{
  "CallbackCommand": "Group.CallbackAfterCreateGroup", // 回调命令
  "GroupId": "@TGS#2J4SZEAE",
  "Operator_Account": "group_root", // 操作者
  "Owner_Account": "leckie", // 群主
  "Type": "Public", // 群组类型
  "Name": "MyFirstGroup", // 群组名称
  "MemberList": [ // 初始成员列表
    {
      "Member_Account": "bob"
    },
    {
      "Member_Account": "peter"
    }
  ],
  "UserDefinedDataList": [ // 用户建群时的自定义字段
    {
      "Key": "UserDefined1",
      "Value": "hello"
    },
    {
      "Key": "UserDefined2",
      "Value": "world"
    }
  ]
}
    
```

```

    ]
  }

```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
GroupId	String	操作的群ID。
Operator_Account	String	发起创建群组请求的操作者ID。
Owner_Account	String	请求创建的群的群主ID。
Type	String	请求创建的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Name	String	请求创建的群组的名称。
MemberList	Array	请求创建的群组的初始化成员列表。
UserDefinedDataList	Array	用户建群时的自定义字段，这个字段默认是没有的，需要开通，详见 <a href="#">自定义字段</a> 。

## 5.7 回调应答包示例

APP后台同步数据后，发送回调应答包。

```

{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略回调结果
}

```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，此处填0表示忽略应答结果。
ErrorInfo	String	必	错误信息。

		填	
--	--	---	--

## 6 参考

[第三方回调简介](#)；

REST API：[创建群组](#)。

# 申请入群之前回调

最近更新时间：2017-03-09 09:13:35

## 1 功能说明

APP后台可以通过该回调实时监控用户的申请加群消息，包括：

1. APP后台可以拦截用户申请加群的操作。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

APP用户通过客户端发起加群申请。

## 4 回调发生时机

用户发起加群申请之后、云通信后台将用户加入群之前进行回调（如果需要管理员审批，则在通知管理员之前）。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackBeforeApplyJoinGroup。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackBeforeApplyJoinGroup", // 回调命令
  "GroupId": "@TGS#2J4SZEAEI",
  "Type": "Public", // 群组类型
  "Requestor_Account": "jared" // 申请者
}
```

### 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
GroupId	String	产生群消息的群组ID。
Type	String	产生群消息的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Requestor_Account	String	申请加群者ID。

### 5.7 回调应答包示例

#### 5.7.1 允许继续处理

允许用户加群请求继续处理。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 表示允许继续处理加群请求。
}
```

### 5.7.2 拒绝请求

不允许加群请求继续处理，用户将不能加群，同时给调用方返回错误码：10016。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 1 // 表示拒绝加群请求。
}
```

## 5.9 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许继续处理；1表示拒绝该请求。如果申请加群需要管理员审批，即使回调返回0，依然需要等待管理员审批。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介。](#)

# 拉人入群之前回调

最近更新时间：2017-03-09 03:15:04

## 1 功能说明

APP后台可以通过该回调实时监控群成员拉其他用户入群请求，包括：

1. APP后台可以拦截群成员直接将其他用户拉入群请求。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发起将其他用户拉入群请求。
2. APP管理员通过REST API添加用户到群组。

## 4 回调发生时机

某一成员发起将其他用户拉入群请求之后、云通信后台将目标用户加入群组之前（如果存在关系链托管，且APP在云通信中配置了好友关系校验，则回调发生在好友关系校验通过之后）。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackBeforeInviteJoinGroup。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackBeforeInviteJoinGroup",
  "GroupId": "@TGS#2J4SZEAE",
  "Type": "Public",
  "Operator_Account": "leckie",
  "DestinationMembers": [
    {
      "Member_Account": "jared"
    },
    {
      "Member_Account": "leckie"
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
GroupId	String	要将其他用户拉入的群组ID。

Type	String	请求创建的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Operator_Account	String	请求的操作者ID。
DestinationMembers	Array	要拉入群组的用户ID集合。

## 5.7 回调应答包示例

### 5.7.1 允许所有用户入群

APP后台同意所有请求中所有用户加入群。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 表示允许继续处理加群请求。
}
```

### 5.7.2 拦截部分成员入群

APP后台拒绝请求中的某些用户被拉入群，在RefusedMembers\_Account中返回这些用户ID。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0,
  "RefusedMembers_Account": [ // 拒绝加入的用户列表
    "jared"
  ]
}
```

## 5.9 应答包字段说明

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

RefusedMembers_Account	Array	选 填	拒绝加入的用户ID集合。
------------------------	-------	--------	--------------

## 6 参考

[第三方回调简介](#)。

REST API：[增加群组成员](#)。

# 新成员入群之后回调

最近更新时间：2017-03-09 02:13:13

## 1 功能说明

APP后台可以通过该回调实时监控群成员加入的消息，包括：

1. 通知APP后台有成员入群，APP可以据此进行必要的同步。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端主动申请入群并得到通过。
2. APP用户通过客户端拉其他人入群成功。
3. APP管理员通过REST API添加用户到群组。

## 4 回调发生时机

用户主动申请入群并得到通过，或者被其他用户邀请入群之后，APP管理员通过REST API添加群成员。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterNewMemberJoin。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterNewMemberJoin", // 回调命令
  "GroupId": "@TGS#2J4SZEAE",
  "Type": "Public", // 群组类型
  "JoinType": "Apply", // 入群方式：Apply（申请入群）；Invited（邀请入群）。
  "Operator_Account": "leckie", // 操作者成员
  "NewMemberList": [ // 新入群成员列表
    {
      "Member_Account": "jared"
    },
    {
      "Member_Account": "tommy"
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。

GroupId	String	要将其他用户拉入的群组ID。
Type	String	请求创建的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
JoinType	String	入群方式：Apply（申请入群）；Invited（邀请入群）。
Operator_Account	String	请求的操作者ID。
NewMemberList	Array	新入群成员ID集合。

## 5.7 回调应答包示例

APP后台同步数据后，返回应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略应答结果
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许忽略应答结果。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)；

REST API：[增加群组成员](#)。

# 群成员离开之后回调

最近更新时间：2017-03-09 08:56:34

## 1 功能说明

APP后台可以通过该回调实时监控用户的退群动态，包括：

1. 对用户退群进行实时记录（例如记录日志，或者同步到其他系统）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端退群；
2. APP用户通过客户端踢人；
3. APP管理员通过REST API删除群成员。

## 4 回调发生时机

用户主动退群，或者被群主/管理员踢出群之后。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterMemberExit。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterMemberExit", // 回调命令
  "GroupId": "@TGS#2J4SZEAE", // 群组ID
  "Type": "Public", // 群组类型
  "ExitType": "Kicked", // 成员离开方式：Kicked-被踢；Quit-主动退群。
  "Operator_Account": "leckie", // 操作者
  "ExitMemberList": [ // 离开群的成员列表
    {
      "Member_Account": "jared"
    },
    {
      "Member_Account": "tommy"
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。

GroupId	String	产生群消息的群组ID。
Type	String	产生群消息的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
ExitType	String	成员离开方式：Kicked为被群主踢出；Quit为主动退群。
Operator_Account	String	退群者ID。
ExitMemberList	Array	退出群的成员列表。

## 5.7 回调应答包示例

APP后台同步数据后，返回应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略应答结果
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许忽略应答结果。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)。

REST API：[删除群组成员](#)。

# 群组满员之后回调

最近更新时间：2017-03-09 11:07:30

## 1 功能说明

APP后台可以通过该回调实时监控群组满员的动态，包括：

1. 删除一部分不活跃的群成员，以确保用户能够加入该群。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端申请加群；
2. APP用户通过客户端邀请加群；
3. APP管理员通过REST API增加群组成员。

## 4 回调发生时机

新成员加入之后导致群组满员，或者由于群组满员导致新成员加入失败之后。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterGroupFull。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterGroupFull", // 回调命令
  "GroupId": "@TGS#2J4SZEAE" // 群组ID
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
GroupId	String	满员的群组ID。

## 5.7 回调应答包示例

APP后台记录群满员信息后，发送回调应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略应答结果
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许忽略应答结果。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)。

REST API：[删除群组成员](#)。

REST API：[增加群组成员](#)。

# 群组解散之后回调

最近更新时间：2017-03-09 02:13:16

## 1 功能说明

APP后台可以通过该回调实时监控群组的解散动态，包括：

1. 对群组的解散实时记录（例如记录日志，或者同步到其他系统）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端解散群组；
2. APP管理员通过REST API解散群组。

## 4 回调发生时机

群组解散之后。

## 5 接口说明

### 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content  
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterGroupDestroyed。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterGroupDestroyed", // 回调命令
  "GroupId": "@TGS#2J4SZEAEI",
  "Type": "Public", // 群组类型
  "Owner_Account": "leckie", // 群主
  "Name": "MyFirstGroup", // 群组名称
  "MemberList": [ // 被解散的群组中的成员
    {
      "Member_Account": "leckie"
    },
    {
      "Member_Account": "peter"
    },
    {
      "Member_Account": "bob"
    }
  ]
}
```

### 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。

GroupId	String	被解散的群组ID。
Type	String	被解散群的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Owner_Account	String	群主ID。
MemberList	Array	被解散群的成员列表。

## 5.7 回调应答包示例

APP后台记录群满员信息后，发送回调应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略应答结果
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许忽略应答结果。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)。

REST API：[解散群组](#)。

# 群组资料修改之后回调

最近更新时间：2017-03-09 02:13:17

## 1 功能说明

APP后台可以通过该回调实时监控群组资料的变更，包括：

1. 对修改群组资料的实时记录（例如记录日志，或者同步到其他系统）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[第三方回调配置](#)；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[第三方回调：安全考虑](#)。

## 3 可能触发该回调的场景

### 3.1 哪些内容会触发回调

群组资料包括[群基础资料](#)和[群组维度自定义字段](#)。

1. 目前，群基础资料中的群组名称、群组简介、群组公告和群组头像URL被修改后，可能触发该回调；其他群基础资料被修改暂后不会触发；
2. 所有的群组维度自定义字段被修改后，可能触发该回调。

### 3.2 哪些方式会触发回调

1. APP用户通过客户端修改群组资料；
2. APP管理员通过REST API修改群组资料。

## 4 回调发生时机

群基础资料修改之后。

## 5 接口说明

## 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&content
type=json&ClientIP=$ClientIP&OptPlatform=$OptPlatform
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Group.CallbackAfterGroupInfoChanged。
contenttype	固定为：json。
ClientIP	客户端IP地址。
OptPlatform	客户端平台。取值参见 <a href="#">回调协议</a> 中OptPlatform的参数含义。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

### 5.5.1 仅修改部分群基础资料

如果仅修改了部分群基础资料，则请求包中只包含被修改了的群基础资料字段。

```
{
  "CallbackCommand": "Group.CallbackAfterGroupInfoChanged", // 回调命令
  "GroupId": "@TGS#2J4SZEAE",
  "Type": "Public", // 群组类型
  "Operator_Account": "leckie", // 操作者
  "Notification": "NewNotification" // 修改后的群公告
}
```

### 5.5.2 仅修改部分群维度自定义字段

如果仅修改了部分群维度自定义字段，则请求包中只包含被修改了的群维度自定义字段。

```

{
  "CallbackCommand": "Group.CallbackAfterGroupInfoChanged", // 回调命令
  "GroupId": "@TGS#2J4SZEAE",
  "Type": "Public", // 群组类型
  "Operator_Account": "leckie", // 操作者
  "UserDefinedDataList": [ // 被修改了的群维度自定义字段
    {
      "Key": "UserDefinedKey2", // 自定义字段的Key
      "Value": "UserDefinedValue2" // 自定义字段的Value
    }
  ]
}
    
```

### 5.5.3 同时修改群基础资料和群维度自定义字段

```

{
  "CallbackCommand": "Group.CallbackAfterGroupInfoChanged", // 回调命令
  "GroupId": "@TGS#2J4SZEAE",
  "Type": "Public", // 群组类型
  "Operator_Account": "leckie", // 操作者
  "Name": "NewGroupName", // 修改后的群名称
  "Introduction": "NewIntroduction", // 修改后的群简介
  "Notification": "NewNotification", // 修改后的群公告
  "FaceUrl": "NewFaceUrl", // 修改后的群头像URL
  "UserDefinedDataList": [ // 被修改了的群维度自定义字段
    {
      "Key": "UserDefinedKey1", // 自定义字段的Key
      "Value": "UserDefinedValue1" // 自定义字段的Value
    },
    {
      "Key": "UserDefinedKey2",
      "Value": "UserDefinedValue2"
    },
    {
      "Key": "UserDefinedKey3",
      "Value": "UserDefinedValue3"
    }
  ]
}
    
```

### 5.6 回调请求包字段说明

字段	类型	说明

CallbackCommand	String	回调命令。
GroupId	String	群资料被修改的群组ID。
Type	String	群资料被修改的群的 <a href="#">群组类型</a> ，例如Private，Public和ChatRoom。
Operator_Account	String	操作者ID。
Name	String	修改后的群名称。
Introduction	String	修改后的群简介。
Notification	String	修改后的群公告。
FaceUrl	String	修改后的群头像URL。
UserDefinedDataList	Array	被修改的群维度自定义字段。

## 5.7 回调应答包示例

APP后台记录群资料的变更信息后，发送回调应答包。

```
{
  "ActionStatus": "OK",
  "ErrorInfo": "",
  "ErrorCode": 0 // 忽略应答结果
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示允许忽略应答结果。
ErrorInfo	String	必填	错误信息。

## 6 参考

---

[第三方回调简介](#)。

REST API：[修改群组基础资料](#)。

# 在线状态相关回调

## 状态变更回调

最近更新时间：2017-11-18 14:39:34

## 1 功能说明

App后台可以通过该回调实时监控用户上、下线的行为，包括：

1. 用户上线（TCP连接建立）；
2. 用户注销下线或者用户网络断开（TCP连接断开）；
3. App心跳超时（App异常杀后台或者Crash）。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见第三方回调配置；
2. 回调的方向是：云通讯后台向APP后台发起HTTP POST请求；
3. App后台在收到回调请求之后，**务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid**；
4. 其他安全相关事宜请参考第三方回调：安全考虑；
5. **可能会有大量的心跳超时包（Crash、杀后台等），业务需注意回调响应服务器性能。**

## 3 可能触发该回调的场景

1. 用户通过客户端发起上线请求；
2. 用户通过客户端发起退出的下线请求；
3. 客户端杀后台进程，云服务检测到客户端网络断开后触发下线回调；
4. 客户端心跳超时，包括客户端Crash、关闭网络400秒后，云服务检测到客户端的心跳超时触发下线回调。

## 4 回调发生时机

IM云server收到客户端的**TCP连接建立、TCP连接断开、若干心跳包**无法收到后发生。

## 5 接口说明

## 5.1 请求URL

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

此处假定APP配置的回调URL为 `https://www.example.com`。

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为： <code>State.StateChange</code> 。
contenttype	固定为： <code>json</code> 。
ClientIP	客户端IP，格式如： <code>127.0.0.1</code> 。
OptPlatform	设备类型，分为Windows，Web，Android，iOS，Mac和Unknown。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "State.StateChange",
  "Info": {
    "Action": "Logout",
    "To_Account": "testuser316",
    "Reason": "Unregister"
  }
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
Info	Object	用户上下线的信息。

To_Account	String	用户的ID。
Action	String	App上线或者下线的动作， Login：上线（TCP建立）， Logout下线（TCP断开）。
Reason	String	App上下线触发的原因。上线的原因有Register:App TCP连接建立。下线的原因有Unregister:App用户注销账号导致TCP断开; LinkClose:云通信检测到App TCP连接断开; Timeout：云通信检测到App心跳包超时，认为TCP已断开（客户端杀后台或Crash）。

## 5.7 回调应答包示例

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

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示APP后台处理成功，1表示APP后台处理失败。
ErrorInfo	String	必填	错误信息。

## 6 参考

[第三方回调简介](#)

[用户在线状态综述](#)

# 关系链系统相关回调

## 添加好友之前回调

最近更新时间：2017-11-18 14:39:52

### 1 功能说明

APP后台可以通过该回调实时分析用户的加好友行为，并对本次加好友行为作出相应的处理，例如：APP后台可以打击恶意加好友行为，禁止恶意用户添加好友。

### 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 如果当前请求是管理员发起的加好友请求，云通信后台不会往APP后台发送添加好友之前的回调请求；
5. 其他安全相关事宜请参考[安全考虑](#)。

### 3 可能触发该回调的场景

APP用户通过客户端发起添加其他用户为好友的请求。

### 4 回调发生时机

用户通过客户端发起加好友请求，云通信后台收到加好友请求后，先往APP后台发送加好友之前的回调请求，如果app后台允许该用户发起加好友请求，则继续后面的加好友处理逻辑。

### 5 接口说明

#### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	回调命令字，本回调中该字段固定为：Sns.CallbackPrevFriendAdd。
contenttype	回调的内容类型，本回调中该字段固定为：json。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Sns.CallbackPrevFriendAdd",
  "Requester_Account": "id",
  "From_Account": "id",
  "AddType": "Add_Type_Both",
  "FriendItem": [
    {
      "To_Account": "id1",
      "Remark": "remark1",
      "GroupName": "同学",
      "AddSource": "AddSource_Type_XXXXXXXX",
      "AddWording": "I'm remark1"
    },
    {
      "To_Account": "id2",
      "Remark": "remark2",
      "GroupName": "同事",
      "AddSource": "AddSource_Type_XXXXXXXX",
      "AddWording": "I'm remark2"
    },
    {
      "To_Account": "id3",
      "Remark": "remark3",
      "GroupName": "家人",
      "AddSource": "AddSource_Type_XXXXXXXX",
      "AddWording": "I'm remark3"
    }
  ]
}
```

```

}
],
"ForceAddFlags":0
}
    
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令字，本回调中该字段固定为： Sns.CallbackPrevFriendAdd。
Requester_Account	String	发起加好友请求的用户的Identifier。
From_Account	String	请求加好友的用户的Identifier，对应加好友请求里面的From_Account。
AddType	String	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。
FriendItem	Array	加好友请求的对象数组。
To_Account	String	From_Account请求添加为好友的账号。
Remark	String	From_Account为To_Account设置的备注信息，详情可参见 <a href="#">标配好友字段</a> 。
GroupName	String	From_Account为To_Account设置的分组信息，详情可参见 <a href="#">标配好友字段</a> 。
AddSource	String	加好友来源，详情可参见 <a href="#">标配好友字段</a> 。
AddWording	String	加好友附言，详情可参见 <a href="#">标配好友字段</a> 。
ForceAddFlags	Integer	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。

## 5.7 回调应答包示例

```

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

```

{
  "To_Account": "id1",
  "ResultCode": 0,
  "ResultInfo": ""
},
{
  "To_Account": "id2",
  "ResultCode": 0,
  "ResultInfo": ""
},
{
  "To_Account": "id3",
  "ResultCode": 38000,
  "ResultInfo": "操作过于频繁，请稍后再添加"
}
]
}
    
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	APP后台处理结果： 0表示APP后台处理成功，后台将根据ResultCode来判断是否继续处理加好友逻辑； 非0表示APP后台处理失败，云通信后台将不判断ResultCode，正常处理加好友逻辑。
ErrorInfo	String	必填	APP后台处理失败时返回给云通信后台的错误信息。
ResultItem	Array	必填	APP后台处理结果的对象数组。
To_Account	String	必填	From_Account请求添加为好友的账号。
ResultCode	Integer	必填	错误码信息： 1、0表示允许添加，非0值表示不允许发起添加好友请求； 2、非0类型的错误码的合理取值范围是：【38000,39000】。
ResultInfo	String	必填	不允许发起添加好友请求的原因。

## 6 参考

- 
- 1、[第三方回调简介](#)；
  - 2、[添加好友](#)。

# 添加好友之后回调

最近更新时间：2017-12-04 12:12:31

## 1 功能说明

APP后台可以通过该回调实时监控好友表的增加情况，包括：

1. APP后台可以修改相应的好友表；
2. APP后台可以修改相应的推荐列表。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发起添加其他用户为好友的请求；
2. APP管理员通过REST API添加好友。

## 4 回调发生时机

用户发起加好友请求，后台加好友成功后。

## 5 接口说明

### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Sns.CallbackFriendAdd。
contenttype	固定为：json。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```
{
  "CallbackCommand": "Sns.CallbackFriendAdd",
  "PairList": [
    {
      "From_Account": "id",
      "To_Account": "id1",
      "Initiator_Account": "id"
    },
    {
      "From_Account": "id",
      "To_Account": "id2",
      "Initiator_Account": "id"
    },
    {
      "From_Account": "id",
      "To_Account": "id3",
      "Initiator_Account": "id"
    }
  ],
  "ClientCmd": "friend_add",
  "Admin_Account": "",
  "ForceFlag": 1
}
```

### 5.6 回调请求包字段说明

字段	类型	说明
----	----	----

CallbackCommand	String	回调命令。
PairList	Array	成功添加的关系链对。
From_Account	String	From_Account的好友表中增加了To_Account。
To_Account	String	To_Account被增加到了From_Account的好友表中。
Initiator_Account	String	发起加好友请求的用户的账号。
ClientCmd	String	触发回调的命令字： 1、加好友请求，合理的取值如下：friend_add、FriendAdd； 2、加好友回应，合理的取值如下：friend_response、FriendResponse。
Admin_Account	String	如果当前请求是后台触发的加好友请求，则该字段被赋值为管理员账号；否则为空。
ForceFlag	Integer	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。

## 5.7 回调应答包示例

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

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示APP后台处理成功，1表示APP后台处理失败。
ErrorInfo	String	必填	错误信息。

## 6 参考

- 1、[第三方回调简介](#)；
- 2、[添加好友](#)。

# 添加好友回应之前回调

最近更新时间：2017-11-18 14:40:21

## 1 功能说明

APP后台可以通过该回调实时分析用户的加好友回应行为，并对本次加好友回应行为作出相应的处理，例如：APP后台可以打击恶意加好友行为，禁止恶意用户添加好友。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 如果当前请求是管理员发起的加好友回应请求，云通信后台不会往APP后台发送添加好友回应之前的回调请求；
5. 其他安全相关事宜请参考[安全考虑](#)。

## 3 可能触发该回调的场景

用户收到加好友请求后，响应该加好友请求。

## 4 回调发生时机

用户通过客户端发起加好友回应请求，云通信后台收到加好友回应请求后，先往APP后台发送加好友回应之前的回调请求，如果app后台允许该用户的加好友回应请求，则继续后面的加好友回应处理逻辑。

## 5 接口说明

### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	回调命令字，本回调中该字段固定为：Sns.CallbackPrevFriendResponse。
contenttype	回调的内容类型，本回调中该字段固定为：json。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```

{
  "CallbackCommand":"Sns.CallbackPrevFriendResponse",
  "Requester_Account":"id",
  "From_Account":"id",
  "ResponseFriendItem":[
    {
      "To_Account":"id1",
      "Remark":"remark1",
      "TagName":"同学",
      "ResponseAction":"Response_Action_Agree"
    },
    {
      "To_Account":"id2",
      "Remark":"remark2",
      "TagName":"同学",
      "ResponseAction":"Response_Action_AgreeAndAdd"
    },
    {
      "To_Account":"id3",
      "Remark":"",
      "TagName":"",
      "ResponseAction":"Response_Action_Reject"
    }
  ]
}
    
```

### 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令字，本回调中该字段固定为： Sns.CallbackPrevFriendResponse。
Requester_Account	String	发起加好友回应请求的用户的Identifier。
From_Account	String	回应加好友请求的用户的Identifier。
ResponseFriendItem	Array	加好友回应请求的对象数组。
To_Account	String	From_Account请求添加为好友的账号。
Remark	String	From_Account为To_Account设置的备注信息，详情可参见 <a href="#">标配好友字段</a> 。
TagName	String	From_Account为To_Account设置的分组信息，详情可参见 <a href="#">标配好友字段</a> 。
ResponseAction	String	对加好友请求的响应动作： Response_Action_Agree - 同意对方加自己为好友； Response_Action_AgreeAndAdd - 同意并添加对方为好友； Response_Action_Reject - 拒绝对方添加自己为好友。

## 5.7 回调应答包示例

```

{
  "ActionStatus": "OK",
  "ErrorCode": 0,
  "ErrorInfo": "",
  "ResultItem": [
    {
      "To_Account": "id1",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id2",
      "ResultCode": 0,
      "ResultInfo": ""
    },
    {
      "To_Account": "id3",
      "ResultCode": 38000,
      "ResultInfo": "操作过于频繁，请稍后再添加"
    }
  ]
}
    
```

```
]
}
```

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	APP后台处理结果： 0表示APP后台处理成功，后台将根据ResultCode来判断是否继续处理加好友回应逻辑； 非0表示APP后台处理失败，云通信后台将不判断ResultCode，正常处理加好友回应逻辑。
ErrorInfo	String	必填	APP后台处理失败时返回给云通信后台的错误信息。
ResultItem	Array	必填	APP后台处理结果的对象数组。
To_Account	String	必填	From_Account请求添加为好友的账号。
ResultCode	Integer	必填	错误码信息： 1、0表示允许回应该加好友请求，非0值表示不允许回应该加好友请求； 2、非0类型的错误码的合理取值范围是：【38000,39000】。
ResultInfo	String	必填	不允许回应添加好友请求的原因。

## 6 参考

- 1、[第三方回调简介](#)；
- 2、[添加好友](#)。

# 删除好友之后回调

最近更新时间：2017-12-04 12:11:32

## 1 功能说明

APP后台可以通过该回调实时监控好友表的增加情况，包括：

1. APP后台可以修改相应的好友表；
2. APP后台可以修改相应的推荐列表。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发起删除好友的请求；
2. APP管理员通过REST API删除好友。

## 4 回调发生时机

用户发起删好友请求，后台删好友成功后。

## 5 接口说明

### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

### 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Sns.CallbackFriendDelete。
contenttype	固定为：json。

### 5.3 HTTP请求方式

POST

### 5.4 HTTP请求包体格式

JSON

### 5.5 回调请求包示例

```
{
  "CallbackCommand": "Sns.CallbackFriendDelete",
  "PairList": [
    {
      "From_Account": "id",
      "To_Account": "id1"
    },
    {
      "From_Account": "id",
      "To_Account": "id2"
    },
    {
      "From_Account": "id",
      "To_Account": "id3"
    }
  ]
}
```

### 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
PairList	Array	成功删除的关系链对。
From_Account	String	From_Account的好友表中删除了To_Account。

To_Account	String	To_Account从From_Account好友表中删除了。
------------	--------	---------------------------------

## 5.7 回调应答包示例

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

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示APP后台处理成功，1表示APP后台处理失败。
ErrorInfo	String	必填	错误信息。

## 6 参考

- 1、[第三方回调简介](#)；
- 2、[删除好友](#)。

# 添加黑名单之后回调

最近更新时间：2017-12-04 12:12:53

## 1 功能说明

APP后台可以通过该回调实时监控黑名单的添加情况，包括：

1. APP后台可以修改相应的好友表；
2. APP后台可以修改相应的黑名单列表；
3. APP后台可以修改相应的推荐列表。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发起把某个用户加入到黑名单的请求；
2. APP用户通过REST API添加黑名单。

## 4 回调发生时机

用户发起加黑名单请求，后台加黑名单成功后。

## 5 接口说明

### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Sns.CallbackBlackListAdd。
contenttype	固定为：json。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Sns.CallbackBlackListAdd",
  "PairList": [
    {
      "From_Account": "id",
      "To_Account": "id1"
    },
    {
      "From_Account": "id",
      "To_Account": "id2"
    },
    {
      "From_Account": "id",
      "To_Account": "id3"
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
PairList	Array	成功添加的黑名单关系链对。

From_Account	String	From_Account的黑名单列表中添加了To_Account。
To_Account	String	To_Account被加入到From_Account的黑名单列表中了。

## 5.7 回调应答包示例

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

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示APP后台处理成功，1表示APP后台处理失败。
ErrorInfo	String	必填	错误信息。

## 6 参考

- 1、[第三方回调简介](#)；
- 2、[添加黑名单](#)。

# 删除黑名单之后回调

最近更新时间：2017-12-04 12:12:00

## 1 功能说明

APP后台可以通过该回调实时监控黑名单的删除情况，包括：

1. APP后台可以修改相应的好友表；
2. APP后台可以修改相应的黑名单列表；
3. APP后台可以修改相应的推荐列表。

## 2 注意事项

1. 要启用回调，必须配置回调URL，并打开本条回调协议对应的开关，配置方法详见[回调配置](#)；
2. 回调的方向是：云通信后台向APP后台发起HTTP POST请求；
3. APP后台在收到回调请求之后，务必校验请求URL中的参数SdkAppid是否是自己的SdkAppid；
4. 其他安全相关事宜请参考[安全考虑](#)。

## 3 可能触发该回调的场景

1. APP用户通过客户端发起把某个用户从黑名单列表中删除的请求；
2. APP用户通过REST API删除黑名单。

## 4 回调发生时机

用户发起删除黑名单请求，后台删除黑名单成功后。

## 5 接口说明

### 5.1 请求URL

假定APP配置的回调URL为 `https://www.example.com`，本回调请求的URL如下：

```
https://www.example.com/SdkAppid=$SdkAppid&CallbackCommand=$CallbackCommand&contenttype=json
```

## 5.2 请求参数

参数	含义
SdkAppid	APP在云通讯申请的Appid。
CallbackCommand	固定为：Sns.CallbackBlackListDelete。
contenttype	固定为：json。

## 5.3 HTTP请求方式

POST

## 5.4 HTTP请求包体格式

JSON

## 5.5 回调请求包示例

```
{
  "CallbackCommand": "Sns.CallbackBlackListDelete",
  "PairList": [
    {
      "From_Account": "id",
      "To_Account": "id1"
    },
    {
      "From_Account": "id",
      "To_Account": "id2"
    },
    {
      "From_Account": "id",
      "To_Account": "id3"
    }
  ]
}
```

## 5.6 回调请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令。
PairList	Array	成功删除的黑名单关系链对。

From_Account	String	From_Account的黑名单列表中删除了To_Account。
To_Account	String	To_Account从From_Account的黑名单列表中删除了。

## 5.7 回调应答包示例

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

## 5.8 应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	请求处理的结果，OK表示处理成功，FAIL表示失败。
ErrorCode	Integer	必填	错误码，0表示APP后台处理成功，1表示APP后台处理失败。
ErrorInfo	String	必填	错误信息。

## 6 参考

- 1、[第三方回调简介](#)；
- 2、[添加黑名单](#)。

# 回调双向认证配置指南

## Apache双向认证配置指南

最近更新时间：2017-03-09 10:40:44

### 1 Apache HTTPS双向认证配置流程

以第三方开发者域名 `www.example.com` 为例。

#### 1.1 情况1：第三方开发者已有权威第三方签发的证书

1. 开发者准备权威第三方为 `www.example.com` 签发的证书 `www.example.com.crt` 和派发的私钥 `www.example.com.key`；注意必须是权威的第三方机构（如天威、globalsign等）签发的证书；
2. 腾讯云给开发者后台提供用于验证请求方（腾讯）证书的CA证书 `TencentQQAuthCA.crt`；
3. 参照 [Apache HTTPS双向认证配置参考](#) 进行配置。

#### 1.2 情况2：第三方开发者向腾讯云申请为域名签发证书

1. 第三方开发者向腾讯云申请为域名签发证书第三方开发者向腾讯云提供开发者后台域名，例如 `www.example.com`；
2. 腾讯云给开发者后台域名 `www.example.com` 签发证书和私钥，`www.example.com.crt` 和 `www.example.com.key`；
3. 腾讯云给开发者后台提供用于验证请求方（腾讯）证书的CA证书 `TencentQQAuthCA.crt`；
4. 参照 [Apache HTTPS双向认证配置参考](#) 进行配置。

### 2 Apache HTTPS双向认证配置参考

将 `www.example.com.crt`、`www.example.com.key` 和 `TencentQQAuthCA.crt` 拷贝到Apache的安装目录的 `conf` 文件夹下。修改 `httpd.conf` 配置文件，参考配置如下。

```
SSLEngine on #开启SSL
SSLCertificateFile "/usr/local/apache2/conf/example.com.crt" #腾讯给第三方签发的证书
SSLCertificateKeyFile "/usr/local/apache2/conf/example.com.key" #和证书配对的私钥
SSLCACertificateFile "/usr/local/apache2/conf/TencentQQAuthCA.crt" #腾讯认证的CA证书
SSLVerifyClient require #验证请求来源
```

# Nginx双向认证配置指南

最近更新时间：2017-03-09 10:54:04

## 1 Nginx HTTPS双向认证配置指南

以第三方开发者域名 `http://www.example.com` 为例。

### 1.1 情况1：第三方开发者已有权威第三方签发的证书

1. 开发者准备权威第三方为 `www.example.com` 签发的证书 `www.example.com.crt` 和派发的私钥 `www.example.com.key`；注意必须是权威的第三方机构（如天威、globalsign等）签发的证书；
2. 腾讯云给开发者后台提供用于验证请求方（腾讯）证书的CA证书 `TencentQQAuthCA.crt`；
3. 参照[Nginx HTTPS双向认证配置参考](#)进行配置。

### 1.2 情况2：第三方开发者向腾讯云申请为域名签发证书

1. 第三方开发者向腾讯云提供开发者后台域名，例如 `www.example.com`；
2. 腾讯云给开发者后台域名 `www.example.com` 签发证书和私钥，`www.example.com.crt` 和 `www.example.com.key`；
3. 同时腾讯云给开发者后台提供用于验证请求方（腾讯）证书的CA证书 `TencentQQAuthCA.crt`；
4. 参照[Nginx HTTPS双向认证配置参考](#)进行配置。

## 2 Nginx HTTPS双向认证配置参考

将 `www.example.com.crt`、`www.example.com.key` 和 `TencentQQAuthCA.crt` 拷贝到Nginx的安装目录的conf文件夹下。修改`nginx.conf`配置文件，参考配置如下。

```
server {
    listen 443 ssl;
    ssl_protocols TLSv1 TLSv1.1;

    server_name www.example.com; #域名
    ssl_certificate www.example.com.crt; #腾讯给第三方签发的证书
    ssl_certificate_key www.example.com.key; #和证书配对的私钥

    ssl_verify_client on; #验证请求来源
    ssl_client_certificate TencentQQAuthCA.crt; #腾讯认证的CA证书

    location / {
```

```
root html;  
index index.html index.htm;  
}  
}
```