

云通信

第三方回调

产品文档



腾讯云

【版权声明】

©2013-2019 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

第三方回调配置

回调命令列表

在线状态相关回调

 状态变更回调

资料关系链相关回调

 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

单聊消息相关回调

 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

 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

 群内发言之前回调

 群内发言之后回调

 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

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

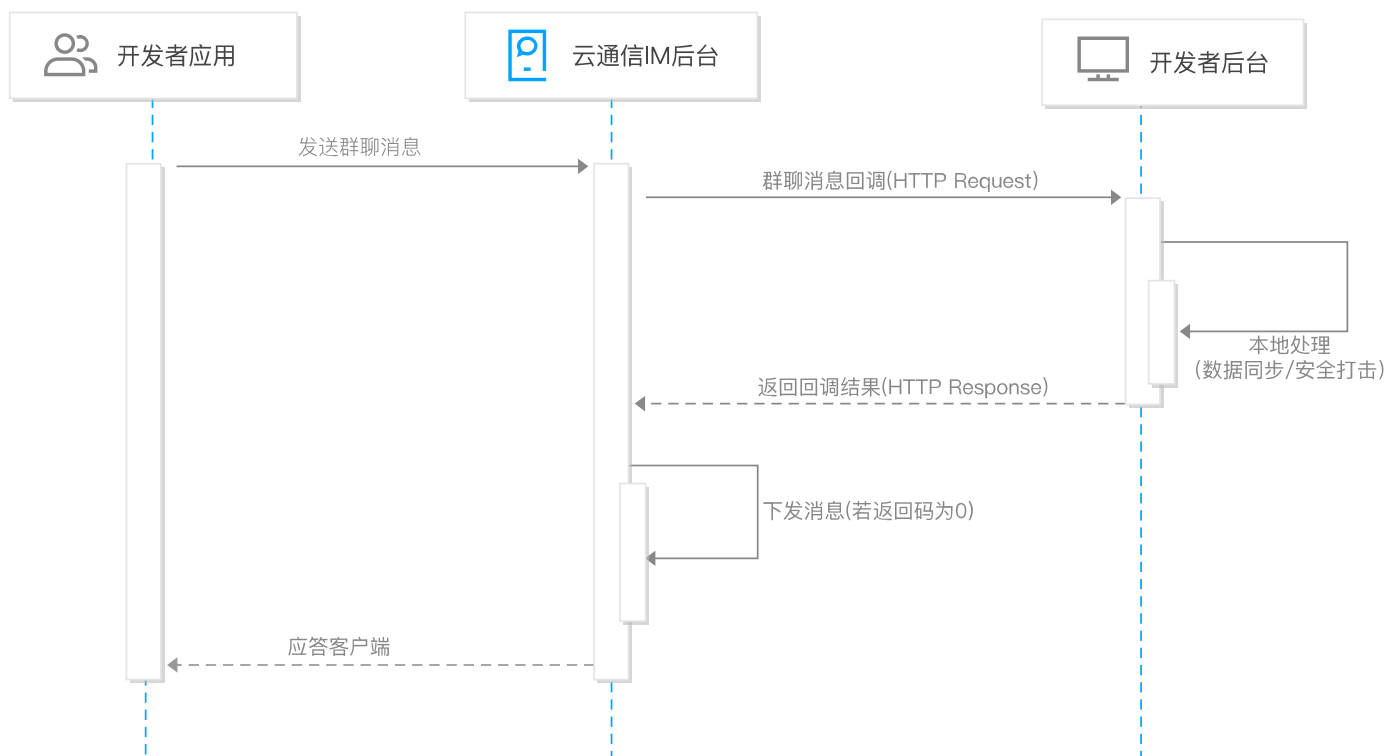
第三方回调简介

最近更新时间：2019-01-29 18:30:07

第三方回调简介

为方便您精细化控制 App 的功能形态，云通信为您提供完全免费且强大的回调能力。所谓回调，即云通信 IM 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](#)。

第三方回调将通过 HTTP/HTTPS 请求的方式发送给 App 后台服务器，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，云通信 IM 后台会在下发该消息之前回调 App 后台服务器，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。回调业务流程如下图所示：



回调分类

从功能角度来看，回调可以分为四大类：

- 在线状态回调
- 关系链相关回调
- 单发消息回调
- 群组相关回调

从处理角度来看，回调可以分为以下两大类：

1. 事件发生之前回调：回调的主要目的在于让 App 后台可以干预该事件的处理逻辑，云通信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 App 后台实现必要的的数据同步，云通信忽略回调返回码（例如群组成员退群之后通知）。

回调协议

第三方回调基于 HTTP/HTTPS 协议，App 后台需要向云通信提供回调 URL，云通信使用 POST 请求的方式来向 App 后台发起回调请求。云通信在发起回调时，会在 App 提供的 URL 之后增加如下几个参数：

参数	含义
SdkAppid	App 在云通信分配的应用标识
CallbackCommand	回调命令字
contenttype	可选，通常值为 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，参见下文回调示例。

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
}
```

回调超时时间

云通信回调 App 后台的超时时间为 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。

安全考虑

云通信支持三种回调类型：

1. HTTP 回调。
2. HTTPS 回调，App 后台的 WebServer 配置的是 CA 机构签发的证书或者是云通信免费签发的证书。

3. HTTPS 双向认证回调，App 后台的 WebServer 配置的是 CA 机构签发的证书或者是云通信免费签发的证书，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云通信。
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云通信。
3. 只有 HTTPS 与双向认证结合，才能确保第三方回调的安全性。

我们强烈建议 App 使用第三种方式实现回调，且云通信签发证书完全免费。

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 [第三方回调配置](#) 文档。

⚠ 注意：

控制台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：

1. 在控制台中配置回调 URL（必须为 HTTPS 域名）、回调开启。
2. 给云通信提需求工单，由云通信给 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](#)

回调不通的常见原因

如果遇到回调不通的情况，App 先依照如下清单排查一下设置的回调服务是否存在问题。

回调不通的现象	可能存在的原因
---------	---------

回调不通的现象	可能存在的原因
回调 URL 访问超时	1. 云通信无法完成 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，云通信无法完成 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。请确认 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。
回调应答包体解析失败	回调请求包体非 JSON 格式。

第三方回调配置

最近更新时间：2019-01-28 18:38:15

我们使用如下案例来介绍接入第三方回调的配置流程：

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【单发聊天消息之前回调】，目的是实时记录 App 中用户的发消息情况。

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云通信控制台自助完成。在云通信 [控制台](#) 中单击对应应用最右侧的【更多】>【回调配置】：

SDKAPPID	应用名称	业务版本	状态	创建时间	到期时间	业务调整	操作
140000983	测试云通信	体验版	启用	2019-01-24 1...	永久生效	升级	应用配置 统计分析 更多
140000727	云通信测试	体验版	启用	2019-01-23 1...	永久生效	升级	应用配置 查看Crash 回调配置 功能配置 开发辅助工具
140000787	测试应用1	IM	启用	2018-06-08 1...			应用配置

进入该应用回调配置页面，如下图：

< 返回 | 测试云通信

基础配置 功能配置 Crash 开发辅助工具

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

- 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
- 修改群名称之后回调
- 修改群介绍之后回调
- 修改群公告之后回调
- 修改群头像URL之后回调

【单发消息之前】和【单发聊天
之前】回调配置完成，将在 5 分

.tencent.com [更改URL](#)

事件回调配置

群组

- 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
- 修改群名称之后回调
- 修改群介绍之后回调
- 修改群公告之后回调
- 修改群头像URL之后回调

资料关系链

- 添加好友之后回调
- 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

单聊消息

- 发单聊消息之前回调
- 发单聊消息之后回调

回调命令列表

最近更新时间：2019-01-28 18:38:39

在线状态

回调类型	回调命令字
状态变更回调	State.StateChange

资料关系链

回调类型	回调命令字
添加好友之前回调	Sns.CallbackPrevFriendAdd
添加好友之后回调	Sns.CallbackFriendAdd
添加好友回应之前回调	Sns.CallbackPrevFriendResponse
删除好友之后回调	Sns.CallbackFriendDelete
添加黑名单之后回调	Sns.CallbackBlackListAdd
删除黑名单之后回调	Sns.CallbackBlackListDelete

单聊消息

回调类型	回调命令字
发单聊消息之前回调	C2C.CallbackBeforeSendMsg
发单聊消息之后回调	C2C.CallbackAfterSendMsg

群组

回调类型	回调命令字
------	-------

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

在线状态相关回调

状态变更回调

最近更新时间：2019-01-28 18:38:46

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：

- 用户上线（TCP 连接建立）
- 用户注销下线或者用户网络断开（TCP 连接断开）
- App 心跳超时（App 异常 kill 后台或者 Crash）

注意事项

- 要启用回调，必须配置回调 URL，并打开本回调对应的开关，配置方法详见 [第三方回调配置](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。
- 可能会有大量的心跳超时包（Crash、kill 后台等），业务需注意回调响应服务器性能。

可能触发该回调的场景

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

回调发生时机

IM 云服务器收到客户端的 TCP 连接建立、TCP 连接断开、若干心跳包无法收到后发生。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：State.StateChange
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

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

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

字段	类型	说明
To_Account	String	用户 Identifier
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； 下线的的原因有 Unregister：App 用户注销帐号导致 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）

应答包示例

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

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

参考

- [第三方回调简介](#)
- [在线状态管理](#)

资料关系链相关回调

添加好友之前回调

最近更新时间：2019-01-28 18:38:50

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，禁止恶意用户发起加好友请求。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

App 用户通过客户端发起加好友请求。

回调发生时机

用户通过客户端发起加好友请求，云通信后台收到该请求后，先往 App 后台发送添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，云通信后台将继续后面的加好友处理逻辑。

接口说明

请求 URL 示例

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

示例：


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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackPrevFriendAdd
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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
}

```

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 请求添加的用户的 Identifier
Remark	String	From_Account 为 To_Account 设置的备注信息，详情可参见 标配好友字段
GroupName	String	From_Account 为 To_Account 设置的分组信息，详情可参见 标配好友字段
AddSource	String	加好友来源，详情可参见 标配好友字段
AddWording	String	加好友附言，详情可参见 标配好友字段
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

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"
}
]
}

```

应答包字段说明

字段	类型	属性	说明
ActionStatus	String	必填	App 后台的处理结果，OK 表示处理成功，FAIL 表示处理失败
ErrorCode	Integer	必填	App 后台处理结果： 0 表示 App 后台处理成功，云通信后台将根据 ResultCode 来判断是否允许加好友 非 0 表示 App 后台处理失败，云通信后台将不再判断 ResultCode，允许该用户加好友
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 请求添加的用户的 Identifier
ResultCode	Integer	必填	错误码信息： 0 表示允许添加，非 0 值表示不允许发起加好友请求 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

参考

- [第三方回调简介](#)
- REST API : [添加好友](#)

添加好友之后回调

最近更新时间：2019-01-28 18:38:55

功能说明

App 后台可以通过该回调实时监控用户的新增好友信息。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 后台通过 REST API 发起加好友请求，请求添加双向好友，且对方的加好友验证方式是“允许任何人”。
- App 用户通过客户端发起加好友请求，请求添加双向好友，且对方的加好友验证方式是“允许任何人”。
- App 后台通过 REST API 发起加好友请求，请求添加单向好友。
- App 用户通过客户端发起加好友请求，请求添加单向好友。
- App 用户收到加好友请求后，同意添加对方为好友。
- App 后台通过 REST API 强制加好友。

回调发生时机

云通信后台收到加好友请求或加好友回应请求，成功添加好友后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackFriendAdd
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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
}
```

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令，本回调中该字段固定为：Sns.CallbackFriendAdd
PairList	Array	成功添加的好友对
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 Identifier
ClientCmd	String	触发回调的命令字： 加好友请求，合理的取值如下：friend_add、FriendAdd 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帐号； 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

应答包示例

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

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

参考

- [第三方回调简介](#)
- REST API : [添加好友](#)

添加好友回应之前回调

最近更新时间：2019-01-28 18:39:00

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。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

App 用户收到加好友请求后，响应该加好友请求。

回调发生时机

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，云通信后台将继续后面的加好友回应逻辑。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackPrevFriendResponse
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

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 请求添加为好友的用户的 Identifier
Remark	String	From_Account 为 To_Account 设置的备注信息，详情可参见 标配好友字段
TagName	String	From_Account 为 To_Account 设置的分组信息，详情可参见 标配好友字段
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

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"
}
]
}
```

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 请求添加为好友的用户的 Identifier
ResultCode	Integer	必填	错误码信息： 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 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

参考

- [第三方回调简介](#)
- REST API：[添加好友](#)

删除好友之后回调

最近更新时间：2019-01-28 18:39:05

功能说明

App 后台可以通过该回调实时监控用户的好友删除信息。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发起删除好友的请求。
- App 后台通过 REST API 发起删除好友的请求。

回调发生时机

云通信后台收到删除好友请求，删好友成功后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackFriendDelete
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令，本回调中该字段固定为：Sns.CallbackFriendDelete
PairList	Array	成功删除的好友
From_Account	String	From_Account 的好友表中删除了 To_Account

字段	类型	说明
To_Account	String	To_Account 从 From_Account 的好友表中删除

应答包示例

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

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

参考

- [第三方回调简介](#)
- REST API：[删除好友](#)

添加黑名单之后回调

最近更新时间：2019-01-28 18:39:08

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。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发起添加黑名单的请求。
- App 后台通过 REST API 发起添加黑名单的请求。

回调发生时机

云通信后台收到加黑名单请求，成功添加黑名单后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackBlackListAdd
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令，本回调中该字段固定为：Sns.CallbackBlackListAdd
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

字段	类型	说明
To_Account	String	To_Account 被加入到 From_Account 的黑名单列表中

应答包示例

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

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

参考

- [第三方回调简介](#)
- REST API : [添加黑名单](#)

删除黑名单之后回调

最近更新时间：2019-01-28 18:39:12

功能说明

App 后台可以通过该回调实时监控用户黑名单的删除情况。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发起删除黑名单请求。
- App 后台通过 REST API 发起删除黑名单请求。

回调发生时机

云通信后台收到删除黑名单请求，成功删除黑名单后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Sns.CallbackBlackListDelete
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令，本回调中该字段固定为：Sns.CallbackBlackListDelete
PairList	Array	成功删除的黑名单对
From_Account	String	From_Account 的黑名单列表中删除了 To_Account

字段	类型	说明
To_Account	String	To_Account 从 From_Account 的黑名单列表中删除

应答包示例

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

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

参考

- [第三方回调简介](#)
- REST API : [添加黑名单](#)

单聊消息相关回调

发单聊消息之前回调

最近更新时间：2019-01-28 18:39:16

功能描述

App 后台可以通过该回调实时监控用户的单聊消息，包括：

- 对发单聊消息进行实时记录（例如记录日志，或者同步到其他系统）。
- 拦截用户的单聊发言请求。
- 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）。

注意事项

- 要启用回调，必须配置回调 URL，并打开本回调对应的开关，配置方法详见 [第三方回调配置](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 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。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发送单聊消息。
- App 管理员通过 REST API（sendmsg 接口）发送单聊消息。

回调发生时机

云通信后台收到用户发送的单聊消息之后、将该消息下发给目标用户之前。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时 在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：C2C.CallbackBeforeSendMsg
contenttype	请求包体固定为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

```
{
  "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"
      }
    }
  ]
}
```

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令
From_Account	String	消息发送者 Identifier
To_Account	String	消息接收者 Identifier
MsgBody	Array	消息体，具体参见 消息格式描述 文档

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
}
```

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
}
```

应答包示例（修改消息内容）

如下的应答示例为：对用户发送的单聊消息进行了修改（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。

示例：

```
{
  "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
}
}
]
}

```

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，云通信后台将把修改后的消息发送给好友，具体格式参见 消息格式描述

参考

- [第三方回调简介](#)
- [发单聊消息之后回调](#)
- REST API：[单发单聊消息](#)
- REST API：[批量发单聊消息](#)

发单聊消息之后回调

最近更新时间：2019-01-28 18:39:19

功能说明

App 后台可以通过该回调实时监控用户的单聊消息，包括：

- 对单聊消息进行实时记录（例如记录日志，或者同步到其他系统）。
- 对单聊消息进行数据统计（例如人数，消息数等）。

注意事项

- 要启用回调，必须配置回调 URL，并打开本回调对应的开关，配置方法详见 [第三方回调配置](#)。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 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。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发送单聊消息。
- App 管理员通过 REST API（sendmsg 接口）发送单聊消息。

回调发生时机

云通信后台收到用户发送的单聊消息并将该消息下发给目标用户之后。

⚠ 注意：

若消息下发失败（例如被脏字过滤），仍会触发该回调。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：C2C.CallbackAfterSendMsg
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

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

请求包字段说明

字段	类型	说明
CallbackCommand	String	回调命令
From_Account	String	消息发送者 Identifier
To_Account	String	消息接收者 Identifier
MsgBody	Array	消息体，具体参见 消息格式描述

应答包示例

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

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

参考

- [第三方回调简介](#)
- [发单聊消息之前回调](#)
- REST API：[单发单聊消息](#)
- REST API：[批量发单聊消息](#)

群组系统相关回调

创建群组之前回调

最近更新时间：2019-01-28 18:39:25

功能说明

App 后台可以通过该回调实时监控用户创建群组的请求，包括后台可以拒绝用户创建群组的请求。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端创建群组
- App 管理员通过 REST API 创建群组

回调发生时机

云通信后台准备创建群组之前。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackBeforeCreateGroup
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

请求包字段说明

对象	介绍	功能
CallbackCommand	String	回调命令
Operator_Account	String	发起创建群组请求的操作者 Identifier
Owner_Account	String	请求创建的群的群主 Identifier

对象	介绍	功能
Type	String	产生群消息的 群组形态介绍 ，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

应答包示例

允许创建

允许用户创建群组。

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

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
}
```

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

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：2019-01-28 18:39:28

功能说明

App 后台可以通过该回调实时监控用户创建群组的信息，包括：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。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端创建群组成功
- App 管理员通过 REST API 创建群组成功

回调发生时机

群组创建成功之后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterCreateGroup
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}

```

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 Identifier
Owner_Account	String	请求创建的群的群主 Identifier
Type	String	请求创建的 群组形态介绍 ，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 自定义字段

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 : [创建群组](#)

申请入群之前回调

最近更新时间：2019-01-28 18:39:32

功能说明

App 后台可以通过该回调实时监控用户申请加群请求，包括：App 后台可以拦截用户申请加群的操作。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

App 用户通过客户端发起加群申请。

回调发生时机

云通信后台将申请加群的用户加入群组之前进行回调（如果需要管理员审批，则在通知管理员之前）。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackBeforeInviteJoinGroup
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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
}
```

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 群组形态介绍 ，例如 Private，Public 和 ChatRoom
Requestor_Account	String	申请加群者 Identifier

应答包示例

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。
}
```

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。
}
```

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

参考

[第三方回调简介](#)

拉人入群之前回调

最近更新时间：2019-01-28 18:39:35

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：App 后台可以拦截群成员直接将其他用户拉入群请求。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

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

回调发生时机

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）。

接口说明

请求 URL 示例

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

示例：

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


请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackBeforeInviteJoinGroup
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

```
{
  "CallbackCommand": "Group.CallbackBeforeInviteJoinGroup",
  "GroupId": "@TGS#2J4SZEAEI",
  "Type": "Public",
  "Operator_Account": "leckie",
  "DestinationMembers": [
    {
      "Member_Account": "jared"
    },
    {
      "Member_Account": "leckie"
    }
  ]
}
```

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 群组形态介绍 ，例如 Private，Public 和 ChatRoom
Operator_Account	String	请求的操作者 Identifier

字段	类型	说明
DestinationMembers	Array	要拉入群组的 Identifier 集合

应答包示例

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。
}
```

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 Identifier。

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

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

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：2019-01-28 18:39:39

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：通知 App 后台有成员入群，App 可以据此进行必要的数据库同步。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

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

回调发生时机

- 用户主动申请并成功加入群组之后。
- 用户被其他成员邀请并成功加入群组之后。
- 用户被App管理员通过 REST API 添加到群组中之后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterNewMemberJoin
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

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	要将其他用户拉入的群组 ID
Type	String	请求创建的 群组形态介绍 ，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 Identifier
NewMemberList	Array	新入群成员 Identifier 集合

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
}
```

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

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：2019-01-28 18:39:44

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：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）。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端退群。
- App 用户通过客户端踢人。
- App 管理员通过 REST API 删除群成员。

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。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterMemberExit
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

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 群组形态介绍 ，例如 Private，Public 和 ChatRoom

字段	类型	说明
ExitType	String	成员离开方式：Kicked 为被群主移出群聊；Quit 为主动退群
Operator_Account	String	退群者 Identifier
ExitMemberList	Array	退出群的成员列表

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
}
```

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

参考

- [第三方回调简介](#)
- REST API：[删除群组成员](#)

群内发言之前回调

最近更新时间：2019-01-28 18:39:48

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：

- 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）。
- 拦截用户在群内发言的请求。
- 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）。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

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

回调发生时机

云通信后台把群消息下发给群成员之前。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackBeforeSendMsg
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
      }
    }
  ]
}
```

请求包字段说明

字段	类型	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 群组形态介绍 ，例如 Private，Public 和 ChatRoom
From_Account	String	消息发送者 Identifier
Operator_Account	String	请求发起者 Identifier，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 消息格式描述

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
}
```

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
}
```

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
}
```

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，例如用户等级、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
      }
    }
  ]
}
```

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，云通信后台将把修改后的消息发送到群组中，具体格式参见 消息格式描述

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：2019-01-28 18:39:52

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

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端发送群消息。
- 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

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterSendMsg
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
      }
    }
  ]
}
```

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 群组形态介绍 ，例如 Private，Public 和 ChatRoom

字段	类型	说明
From_Account	String	消息发送者 Identifier
MsgTime	Integer	消息发送的时间戳，对应后台 Server 时间
MsgSeq	Integer	消息序列号，一条消息的唯一标示
MsgBody	Array	消息体，具体参见 消息格式描述

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](#)

群组满员之后回调

最近更新时间：2019-01-28 18:39:56

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：删除一部分不活跃的群成员，以确保用户能够加入该群。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端申请加群。
- App 用户通过客户端邀请加群。
- App 管理员通过 REST API 增加群组成员。

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。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterGroupFull
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

```
{
  "CallbackCommand": "Group.CallbackAfterGroupFull", // 回调命令
  "GroupId": "@TGS#2J4SZEAEI" // 群组 ID
}
```

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

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
}
```

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

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：2019-01-28 18:40:00

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：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）。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

- App 用户通过客户端解散群组。
- App 管理员通过 REST API 解散群组。

回调发生时机

群组解散之后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterGroupDestroyed
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

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"
    }
  ]
}
```

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	被解散的群组 ID
Type	String	被解散群的 群组形态介绍 ，例如 Private，Public 和 ChatRoom
Owner_Account	String	群主 Identifier
MemberList	Array	被解散群的成员列表

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
}
```

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

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：2019-01-28 18:40:08

功能说明

App 后台可以通过该回调实时监控群组资料（群名称、群简介、群公告、群头像及群维度自定义字段）的变更，包括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）。

注意事项

- 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](#) 文档。
- 回调的方向是云通信后台向 App 后台发起 HTTP POST 请求。
- 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。
- 其他安全相关事宜请参考 [第三方回调简介：安全考虑](#) 文档。

可能触发该回调的场景

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](#)。

- 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；
- 所有的群组维度自定义字段被修改后，可能触发该回调。

哪些方式会触发回调

- App 用户通过客户端修改群组资料。
- App 管理员通过 REST API 修改群组资料。

回调发生时机

群基础资料修改之后。

接口说明

请求 URL 示例

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

示例：

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

请求参数说明

参数	说明
https	请求协议为 HTTPS，请求方式为 POST
www.example.com	回调 URL
SdkAppid	创建应用时在云通信控制台分配的 SdkAppid
CallbackCommand	固定为：Group.CallbackAfterGroupInfoChanged
contenttype	固定值为：json
ClientIP	客户端 IP，格式如：127.0.0.1
OptPlatform	客户端平台，取值参见 第三方回调简介：回调协议 中 OptPlatform 的参数含义

请求包示例

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
}
```

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
    }
  ]
}
```

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"
    }
  ]
}
```

请求包字段说明

字段	类型	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 群组形态介绍 ，例如 Private，Public 和 ChatRoom
Operator_Account	String	操作者 Identifier
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

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
}
```

应答包字段说明

对象	介绍	功能	功能
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

参考

- [第三方回调简介](#)

- REST API : [修改群组基础资料](#)

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：2019-01-28 18:40:12

Apache HTTPS 双向认证配置流程

以第三方开发者域名 `www.example.com` 为例，有如下两种情况：

• 第三方开发者已有权威第三方签发的证书

- 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。
- 云通信给开发者后台提供用于验证请求方（云通信）证书的 CA 证书 [TencentQCAuthCA.crt](#)。
- 参照下文 Apache HTTPS 双向认证配置参考进行配置。

• 第三方开发者向云通信申请为域名签发证书

- 开发者向云通信提供开发者后台域名，例如 `www.example.com`。
- 云通信给开发者后台域名 `www.example.com` 签发证书和私钥，`www.example.com.crt` 和 `www.example.com.key`。
- 云通信给开发者后台提供用于验证请求方（云通信）证书的 CA 证书 [TencentQCAuthCA.crt](#)。
- 参照下文 Apache HTTPS 双向认证配置参考进行配置。

Apache HTTPS 双向认证配置参考

1. 将 `www.example.com.crt`、`www.example.com.key` 和 `TencentQCAuthCA.crt` 拷贝到 Apache 的安装目录的 `conf` 文件夹下。
2. 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/TencentQCAuthCA.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：2019-01-28 18:40:16

Nginx HTTPS 双向认证配置流程

以第三方开发者域名 `http://www.example.com` 为例，有如下两种情况：

• 第三方开发者已有权威第三方签发的证书

- 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。
- 云通信给开发者后台提供用于验证请求方（云通信）证书的 CA 证书 [TencentQQAuthCA.crt](#)。
- 参照下文 Ngin x HTTPS 双向认证配置参考进行配置。

• 第三方开发者向云通信申请为域名签发证书

- 开发者向云通信提供开发者后台域名，例如 `www.example.com`。
- 云通信给开发者后台域名 `www.example.com` 签发证书和私钥，`www.example.com.crt` 和 `www.example.com.key`。
- 云通信给开发者后台提供用于验证请求方（云通信）证书的 CA 证书 [TencentQQAuthCA.crt](#)。
- 参照下文 Ngin x HTTPS 双向认证配置参考进行配置。

Nginx HTTPS 双向认证配置参考

1. 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。
2. 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;
    }
}
```

```
}  
}
```