

# 滔滔对讲 SDK 说明

v3.3.1



北京滔达科技有限公司

[www.allptt.com](http://www.allptt.com)

[13301168635\(vx\)](tel:13301168635)

邮箱: [kylindev@qq.com](mailto:kylindev@qq.com)

## 目录

1 功能.....	1
2 Android 版 SDK 集成.....	1
2.1 概述.....	1
2.2 集成步骤.....	2
2.3 接口说明.....	3
2.3.1 InterpttService 接口 API.....	4
2.3.2 InterpttService 回调 Callback.....	5
2.3.3 设置通知栏图标.....	7
2.3.4 设置悬浮窗图片.....	8
2.3.5 起麦、放麦提示音量设置.....	9
3 iOS 版 SDK 集成.....	9
3.1 概述.....	9
3.2 集成步骤.....	10
3.3 接口说明.....	10
3.3.1 连接管理接口 API.....	10
3.3.2 服务器回调 Callback.....	10
3.3.3 服务器接口 API.....	12
3.3.4 声音接口 API.....	16
3.3.5 声音回调 Callback.....	17
3.3.6 蓝牙手咪接口 API.....	17
3.3.7 蓝牙手咪回调 Callback.....	18
4 web 版集成.....	18
4.1 概述.....	18
4.2 集成步骤.....	19

4.2.1 服务器部署.....	19
4.2.2 浏览器访问验证.....	19
4.3 接口说明.....	20
4.3.1 登录.....	20
4.3.2 频道.....	20
4.3.3 用户.....	21
4.3.4 对讲.....	21
4.3.5 收发即时通信消息.....	22
4.3.6 接收视频直播.....	23
4.3.7 推送文字播报.....	24
4.3.8 收发公告.....	24
4.3.9 查询离线消息（历史消息查询）.....	25
4.3.10 发起音视频通话.....	25
4.3.11 应答音视频通话.....	26
5 后台管理接口.....	27
5.1 概述.....	27
5.1.1 后台业务规则.....	27
5.1.2 返回码（code）.....	33
5.2 系统管理员接口说明（/admin/）.....	33
5.2.1 系统管理.....	33
5.2.1.1 登录（login）.....	34
5.2.1.2 修改账号密码（setAccountPassword）.....	34
5.2.1.3 获取系统配置（getSystemConfig）.....	35
5.2.1.4 修改系统配置（setSystemConfig）.....	36
5.2.2 组织管理（创建、删除、获取、修改密码）.....	37
5.2.2.1 新建组织（createEnt）.....	38
5.2.2.2 删除组织（deleteEnt）.....	38

5.2.2.3 获取组织总数 (getEntsTotal)	39
5.2.2.4 分页获取组织列表 (getEntsInPage)	39
5.2.2.5 重置组织密码 (resetEntPassword)	40
5.2.2.6 获取组织属性 (getEntProperty)	41
5.2.2.7 修改组织属性 (setEntProperty)	42
5.3 组织管理员接口说明 (/ent/)	43
5.3.1 组织管理	43
5.3.1.1 登录 (login)	44
5.3.1.2 修改账号密码 (setAccountPassword)	45
5.3.1.3 获取组织属性 (getEntProperty)	45
5.3.1.4 修改组织属性 (setEntProperty)	47
5.3.1.5 推送通知 (pushNotice)	48
5.3.2 用户管理 (创建、删除、获取、修改)	49
5.3.2.1 新建用户 (createUser)	49
5.3.2.2 批量新建用户 (createUsersInBatch)	50
5.3.2.3 删除用户 (deleteUser)	51
5.3.2.4 获取用户总数 (getUsersTotal)	52
5.3.2.5 分页获取用户列表 (getUsersInPage)	52
5.3.2.6 重置用户密码 (resetUserPassword)	54
5.3.2.7 获取用户属性 (getUserProperty)	54
5.3.2.8 修改用户属性 (setUserProperty)	55
5.3.2.9 所有用户批量增加永久卡 (batchAddPerm)	57
5.3.2.10 所有用户批量增加月卡 (batchAddMonth)	57
5.3.2.11 所有用户补足若干月内服务 (batchAddToMonth)	58
5.3.2.12 分页获取用户上下线记录 (getLoginRecordInPage)	59
5.3.2.13 查询用户指定日期内使用统计 (getUserActivity)	60
5.3.3 频道管理 (创建、删除、获取、修改)	61
5.3.3.1 新建频道 (createChannel)	61
5.3.3.2 删除频道 (deleteChannel)	62
5.3.3.3 获取频道总数 (getChannelsTotal)	62
5.3.3.4 分页获取频道列表 (getChannelsInPage)	63
5.3.3.5 修改频道名称 (setChannelName)	64
5.3.3.6 获取频道属性 (getChannelProperty)	64
5.3.3.7 修改频道属性 (setChannelProperty)	65
5.3.4 频道成员管理 (添加、移除、获取、修改)	66
5.3.4.1 添加频道成员 (addMember)	66
5.3.4.2 移除频道成员 (removeMember)	67
5.3.4.3 获取频道成员总数 (getMembersTotal)	68
5.3.4.4 分页获取频道成员列表 (getMembersInPage)	69
5.3.4.5 获取频道成员身份 (getMemberRole)	69

5.3.4.6 修改频道成员身份 (setMemberRole) .....	70
5.3.4.7 获取频道成员麦权级别 (getMemberLevel) .....	71
5.3.4.8 修改频道成员麦权级别 (setMemberLevel) .....	72
5.3.5 录音及消息管理.....	72
5.3.5.1 分页获取录音列表 (getVoicesInPage) .....	72
5.3.5.2 打包下载录音文件 (downloadVoiceZip) .....	74
5.3.5.3 分页获取频道消息列表 (getMessagesInPage) .....	75
5.3.6 运营统计.....	76
5.3.6.1 频道人数排行 (getChannelRanks) .....	76
5.3.6.2 24 小时内每小时在线人数 (getOnlineNumbers) .....	77
5.3.6.3 最近若干天每日新增人数 (getIncreaseNumbers) .....	78
5.3.6.4 24 小时内登录人数 (getActiveNumber) .....	78
5.3.6.5 当前正在呼叫人数 (getCallingNumber) .....	79
5.3.6.6 当前正在频道内接收语音路数 (getReceivingNumber) .....	80
5.3.6.7 监听语音路数 (getListeningNumber) .....	80
5.3.6.8 最近若干天每日呼叫总次数 (getCallNumbers) .....	81
5.3.6.9 最近若干天每日呼叫总时长 (getCallDurations) .....	82
5.3.6.10 最近若干天每日活跃用户 (getMultiActive) .....	83
5.4 通用接口说明 (/system/) .....	84
5.4.1 获取 token 的账号名 (getAccountByToken) .....	84
5.4.2 获取加密数据的公钥 (getPublicKey) .....	84
6 用户位置及轨迹查询接口.....	85

## 1 功能

滔滔对讲是北京滔达科技有限公司开发的一款智能手机上的语音对讲软件。对讲 SDK 封装了对讲的底层功能，供第三方 app 快速实现语音对讲功能。SDK 实现了与对讲服务器之间的信令处理、语音编解码、蓝牙手咪通信、后台服务运行、耳机线控等事件的监听和处理等功能。

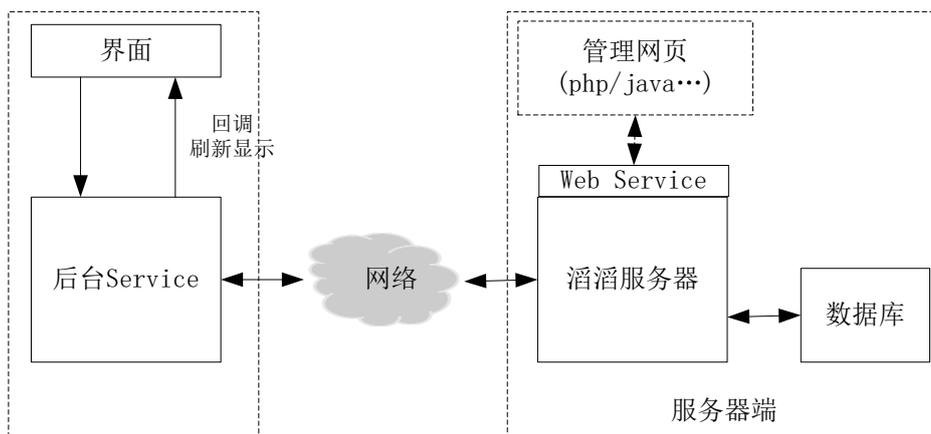
另外，滔达科技还提供 Demo 程序，其中集成了对讲 SDK，并在界面里实现了用户注册、创建/删除频道、频道增加/删除成员、申请话筒、讲话、收听、手咪通信等操作。第三方开发者可参考 Demo 程序，快速实现自己的 app 界面。

因 SDK 可能经常升级，实际集成时如遇文档和 Demo 不一致，以最新 Demo 为准。

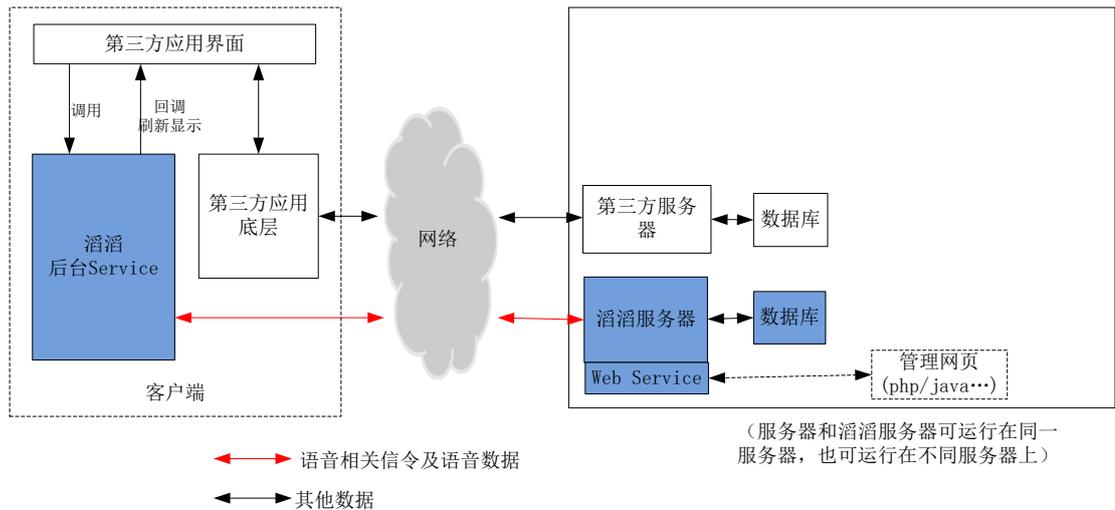
## 2 Android 版 SDK 集成

### 2.1 概述

滔滔对讲 Android 版客户端与后台间的架构如下图：



第三方应用集成对讲 SDK 时，架构如下：



对讲部分的数据库，主要存储用户列表、频道列表，其中频道数据里包括该频道的成员列表。集成时，第三方应用应把群组 and 用户信息同步至对讲服务器。典型方式是平台侧同步，具体见下文“服务器管理接口”部分。

也可以双方服务器无需对接，所有可触发对接数据库增删改查的操作，均通过客户端调用 SDK 实现。例如，客户端新用户注册成功后，应调用 SDK 的 register() 接口，在滔滔数据库里创建该用户。又如，客户端创建了一个群组，可调用 SDK 的 createChannel() 接口，在滔滔数据库里创建对应频道。

## 2.2 集成步骤

SDK 用 Android Studio 开发，具体形式是一个 .aar 文件。集成时，把 ptllib.aar 文件拖入 app 的 libs 目录下即可。在 app 的 build.gradle 文件里，确保有：

```
repositories{
    flatDir{
        dirs 'libs' //就是你放 aar 的目录地址
    }
}
```

另外，在 dependencies 部分，做如下添加：

```
dependencies {
    <工程其他依赖>
    compile(name:'ptllib-release', ext:'aar')
}
```

在 app 的 AndroidManifest.xml 文件里，在 application 下级、与 activity 并列，添加以下内容：

```
<!-- AlarmAlert 是被单呼时，本地弹窗 -->
<activity
    android:name="com.kylindev.ptllib.view.AlarmAlert"
    android:launchMode="singleInstance"
    android:excludeFromRecents="true"
    android:taskAffinity=""
```

```
    android:theme="@android:style/Theme.Wallpaper.NoTitleBar"
    android:configChanges="orientation|keyboardHidden|keyboard|navigation"
    android:screenOrientation="portrait" />

<!-- 用于开机自启动 -->
<receiver android:name="com.kylindev.totalk.utils.BootCompleteReceiver">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
    </intent-filter>
</receiver>

<!-- ptt 功能主服务 -->
<service
    android:name="com.kylindev.pttlib.service.InterpttService"
    android:exported="true" />
</service>

<!-- 监听耳机按键 -->
<receiver
    android:name="com.kylindev.pttlib.utils.MediaButtonIntentReceiver"
    android:enabled="true"
    android:priority="1000">
    <intent-filter>
        <action android:name="android.intent.action.MEDIA_BUTTON" />
    </intent-filter>
</receiver>

<!-- -->
<receiver android:name="com.kylindev.pttlib.utils.AutoRestartServiceReceiver">
    <intent-filter>
        <action android:name="com.kylindev.pttlib.auto_restart_service" />
    </intent-filter>
</receiver>
```

App 在混淆时，应在 proguard-rules.pro 文件里，增加：

```
-dontwarn com.kylindev.pttlib.**
-keep class com.kylindev.pttlib.**{*;} 
```

## 2.3 接口说明

注意：因客户端功能经常更新优化，本文档仅列出部分接口，实际集成已 Demo 程序及代码为准。

SDK 的核心功能由一个始终后台运行的 service 实现，名为 InterpttService。App 在使用 SDK 功能之前，首先应 startService 或 bindService，并在 onServiceConnected() 回

调中获取到 InterpttService 的实例 (mService)。之后, 所有与对讲有关的操作、设置, 均通过调用 mService 的接口完成。

此外, 当 service 接收到服务器数据、或本地音频、蓝牙外设等模块状态改变时, 会发送回调事件。App 可监听这些事件, 刷新界面显示。

以一个 Activity 为例, 首先应 import 如下文件:

**import** com.kylindev.pttlib.service.PttService; (调用对讲 SDK 功能, 必选)

**import** com.kylindev.pttlib.service.PttServiceObserver; (对讲 SDK 回调, 可选)

**import** com.kylindev.pttlib.utils.ServerProto; (如果 Activity 里有与对讲服务器通信相关的功能, 可选)

在 onCreate 里, 首先判断 service 是否已经运行。如果没有, 则先 startService()。之后, 调用 bindService(), 这将触发随后的 onServiceConnected() 被调用, 在其中, 可获得 InterpttService 的实例。更细节的方法请参考 demo 程序代码。

## 2.3.1 InterpttService 接口 API

InterpttService 里可用接口包括 (部分示例, 全部接口请见 demo 代码):

功能	接口	说明
注册新用户	<b>public void</b> register(String host, String id, String pwd, String nickname)	参数依次表示服务器地址 (IP 或 url)、用户名 (字母或数字或组合)、密码 (如果对讲服务器无须检验密码, 可写入固定值, 例如 "123456")、昵称 (如果无须根据对讲显示昵称, 可写入非空固定值)。
用户登录	<b>public void</b> login(String host, String id, String pwd)	参数依次表示服务器地址 (IP 或 url)、用户名 (字母或数字均可)、密码 (如果对讲服务器无须检验密码, 可写入固定值, 例如 "123456")。
创建频道	<b>public void</b> createChannel( <b>final</b> String name, <b>final</b> String pwd, <b>final</b> String members, <b>final boolean</b> searchable, <b>final boolean</b> tmp)	参数说明: name: 频道名称 pwd: 频道口令。Null 表示无口令 members: 用户号以逗号分割, 如 "1001001, 1001002, 1001003"。Null 表示不添加初始成员 searchable: 其他用户是否能搜索到 tmp: 是否是临时频道
加入频道	<b>public void</b> joinChannel( <b>final int</b> channelId, <b>final</b> String pwd)	channelId: 频道号 pwd: 频道口令。如果无需口令, 则传入 null 注意: 加入频道表示成为频道的一个成员, 进入频道 (enterChannel) 才表示真正进入频道、并可与频道内人员开始对讲
进入频道	<b>public void</b> enterChannel( <b>final int</b> channelId)	
申请话筒	<b>public void</b> userPressDown( <b>boolean</b>	fromHeadset: 表示是否由耳机或蓝牙外设触发。由第三方 app 手机界面发起的申

	fromHeadset)	请, 应统一传入 false
归还话筒	<b>public void</b> userPressUp()	

注意: 客户端与服务器之间基于长连接, 与常见的 http 协议 request/response 不同, 请求和回应都是无状态的。以创建频道为例, 客户端向服务器发送 createChannel 消息, 然后无需等待。服务器处理如果成功, 会主动向客户端发送一个 ChannelState 消息; 如果失败, 会向发送一个错误信息。

### 2.3.2 InterpttService 回调 Callback

为接收 service 主动回调, 应在 Activity 里实现一个 BaseServiceObserver 实例。例如:

```
private BaseServiceObserver serviceObserver = new BaseServiceObserver () {
    public void onConnectionStateChanged(int state) throws RemoteException {
    }
    @Override
    public void onPermissionDenied(String reason, int denyType) throws RemoteException {
    }
    @Override
    public void onCurrentChannelChanged() throws RemoteException {
    }
    . . . . .
};
```

在获取到 InterpttService 的实例 mService 后, 应调用:

```
mService.registerObserver(serviceObserver);
```

注册此 Observer。之后即可随时接收 PttService 的回调事件, 并进行相应处理。在 Activity 的 onDestroy 里, 应调用:

```
mService.unregisterObserver(serviceObserver);
```

InterpttService 回调接口包括:

//服务器下发了新的、本地用户可进入的频道:

```
public void onChannelAdded(final Channel channel) throws RemoteException {
}
```

//频道移除了。移除可能是创建者删除了频道, 或者自己退出了该频道, 不再是该频道的成员, 所以本地无需再显示了

```
public void onChannelRemoved(final Channel channel) throws RemoteException {
}
```

//频道资料有变化, 最常见的场景是频道名称改变了。注意: 频道里在线用户变化不在此体现

```
public void onChannelUpdated(final Channel channel) throws RemoteException {
}
```

//与服务器之间的连接状态改变了, 包括掉线、上线、正在同步等

```
public void onConnectionStateChanged(final ConnState state) throws RemoteException {
}
```

//自己所在的频道改变了。例如之前在频道 1, 现在进入了频道 2

```
public void onCurrentChannelChanged() throws RemoteException {
}
//自己的资料改变了，如昵称、头像
public void onCurrentUserUpdated() throws RemoteException {
}
//某个本地可见的用户资料改变了
public void onUserUpdated(final User user) throws RemoteException {
}
//某个本地可见的用户的讲话状态改变了。状态有两种，在讲话，不在讲话
public void onUserTalkingChanged(final User user, boolean talk) throws RemoteException {
}
//某个本地用户的讲话状态改变了。与上一接口的区别在于，onUserTalkingChanged 根据服务器同步下来的信息判断，onLocalUserTalkingChanged 根据本地是否真正有该用户的声音正在播放判断。如果跟该 user 在同一频道，则应根据 onLocalUserTalkingChanged 判断，更加准确、实时。
public void onLocalUserTalkingChanged(final User user, boolean talk) throws RemoteException {
}
//本地播放的实时音量数据。如果不显示，可忽略此接口
public void onNewVolumeData(final short volume) throws RemoteException {
}
//服务器拒绝了某操作，例如试图进入无权限的频道、试图创建超过 3 个频道等
public void onPermissionDenied(String reason, int denyType) throws RemoteException {
}
//注册结果。如果 returnId>0，表明注册成功，且用户号就是 returnId。如果 returnId<=0，表明注册失败
public void onRegisterResult(int returnId) throws RemoteException {
}
public void onForgetPasswordResult(boolean suc) throws RemoteException {
}
public void onRejected(ServerProto.Reject.RejectType type) throws RemoteException {
}
//话筒状态变化。注意，线控、蓝牙触发对讲时，由 service 直接处理。并通过此回调，告知界面此时的话筒状态，包括空闲、申请中、讲话中、归还中
public void onMicStateChanged(MicState s) throws RemoteException {
}
//蓝牙音频连接状态
public void onHeadsetStateChanged(PttService.HeadsetState s) throws RemoteException {
}
//蓝牙 sco 连接状态
public void onScoStateChanged(int s) throws RemoteException {
}
//滔滔手咪连接状态。这个是 BLE 协议的连接
public void onTargetHandmicStateChanged(BluetoothDevice device, PttService.HandmicState s)
throws RemoteException {
}
```

//正在搜索手咪

```
public void onLeDeviceScanStarted(boolean start) throws RemoteException {
}
public void onLeDeviceFound(BluetoothDevice device) throws RemoteException {
}
//讲话计时
public void onTalkingTimerTick(int seconds) throws RemoteException {
}
public void onTalkingTimerCanceled() throws RemoteException {
}
public void onUserAdded(final User u) throws RemoteException {
}
public void onUserRemoved(final User u) throws RemoteException {
}
public void onChannelSearched(int chanId, String chanName, boolean joined, boolean needPwd)
throws RemoteException {
}
public void onShowToast(final String str) throws RemoteException {
}
public void onPlaybackChanged(final int index, final boolean start) throws RemoteException {
}
public void onRecordFinished() throws RemoteException {
}
public void onInvited(final Channel chan) throws RemoteException {
}
public void onUserSearched(User user) throws RemoteException {
}
public void onApplyContactReceived(final boolean add, final Contact contact) throws
RemoteException {
}
public void onPendingContactChanged() throws RemoteException {
}
public void onContactChanged() throws RemoteException {
}
public void onSynced() throws RemoteException {
}
```

### 2.3.3 设置通知栏图标

对讲 SDK 运行时，会在通知栏显示图标及文字。第三方 app 可以设置自己的 icon 进行显示。图标尺寸均建议 96\*96。如果不设置，则显示滔滔默认图标。接口如下：

```
//设置已连接、但所在频道无人讲话时的图标
public void setBitmapNotifConnected(Bitmap bm);
```

```
//设置未连接时的图标  
public void setBitmapNotifDisconnected(Bitmap bm)
```

```
//自己讲话时的图标  
public void setBitmapNotifITalking(Bitmap bm)
```

```
//别人讲话时的图标  
public void setBitmapNotifOtherTalking(Bitmap bm)
```

## 2.3.4 设置悬浮窗图片

对讲 SDK 运行时，如果开关打开，会在程序后台运行时，在桌面显示对讲悬浮窗。第三方 app 可以设置自己的图片进行显示。如果不设置，则显示滔滔默认图标。

悬浮窗的按键，由两张图片上下叠加而成。背景图片（demo 程序里的黄色外圈）表示话筒状态，按钮图片（demo 程序悬浮窗里的灰色部分）表示 ptt 是松开还是按下状态。

两张图片，尺寸均应 >100dp 且尺寸相同。按钮图片的实际显示区域，应与图片矩形边框有一定距离，以使背景显示出来，形成圆环效果。

具体接口如下：

```
//ptt 按钮未就绪时的背景图片。未就绪表示掉线或不在频道里  
//如果不调用此接口，会尝试读取下面 setBitmapFloatWindowPttBackgroundReady 指定的  
图片作为 NoReady 图片。
```

```
public void setBitmapFloatWindowPttBackgroundNoReady(Bitmap bm)
```

```
//ptt 按钮就绪、且所在频道无人讲话时的背景图片  
public void setBitmapFloatWindowPttBackgroundReady(Bitmap bm)
```

```
//正在申请话筒时的背景图片  
public void setBitmapFloatWindowPttBackgroundApplying(Bitmap bm)
```

```
//正在打开蓝牙的背景图片。如果连接了蓝牙音频、且抢麦动作由蓝牙信号或耳机按键触发，  
则 SDK 会自动尝试打开蓝牙 mic 输入。这个过程需要一定时间。再次期间，背景有相应的  
显示
```

```
public void setBitmapFloatWindowPttBackgroundOpeningSco(Bitmap bm)
```

```
//自己正在讲话的背景图片  
public void setBitmapFloatWindowPttBackgroundTalking(Bitmap bm)
```

```
//松开状态的 ptt 按钮  
public void setBitmapFloatWindowPttPicUp(Bitmap bm)
```

```
//按下状态的 ptt 按钮  
public void setBitmapFloatWindowPttPicDown(Bitmap bm)
```

## 2.3.5 起麦、放麦提示音量设置

SDK 的起麦、放麦提示音，默认最大值 100。第三方可设置提示音量，参数范围 0-100。设置 0 时无声，100 时最大声音。

//按下 ptt 键的瞬间

```
public void setVolumeTalkroomPress(int v)
```

//自己起麦（申请话筒成功、开始讲话）

```
public void setVolumeTalkroomBegin(int v)
```

//自己放麦（自己结束讲话）

```
public void setVolumeTalkroomEnd(int v)
```

//别人起麦（别人开始讲话）

```
public void setVolumeOtherBegin(int v)
```

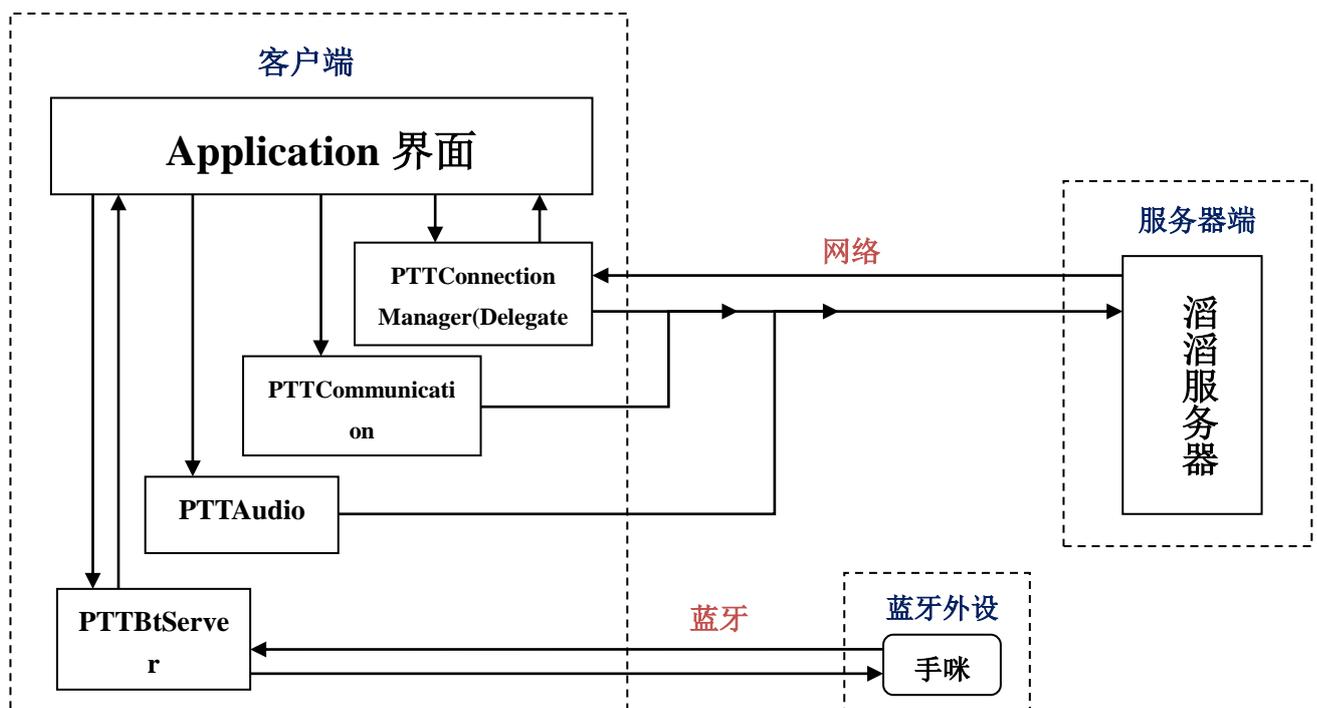
//别人放麦（结束讲话）

```
public void setVolumeOtherEnd(int v)
```

## 3 iOS 版 SDK 集成

### 3.1 概述

iOS SDK 主要接口类关系如下图：



## 3.2 集成步骤

iOS 版 SDK 包括 .a 文件和 .h 文件, 适用于 Xcode 开发环境, 集成时, 把 PTT\_SDK 文件夹 (包含 .a 和 .h 文件) 拖入 app 的工程目录下即可。

info.plist 的 background mode 里, 需配置 voip。

## 3.3 接口说明

### 3.3.1 连接管理接口 API

管理连接、断开等基本事件 PTTConnectionManager  
(PTTConnectionManager.h)

//1、建立连接

```
- (void) connectToHost:(NSString *)host           //服务器地址
    andClientType:(int)clientType                //客户端类型, 0-个人版; 1-企业版
    andUserId:(NSString *)userId                //用户名
    andPassword:(NSString *)password            //密码
    andNick:(NSString *)userNick               //昵称
    andAuthType:(int)authType                  //用于区别是注册还是登录, 0 为注册, 1 为登录
    withParentViewController:(UIViewController *)parentViewController;
```

//2、断开连接

```
- (void) disconnectFromServer;
```

//3、获取连接对应的通信管理器

```
- (PTTCommunicationManager *) getCommunicationManager;
```

//兼容 Swift, 功能同上

```
- (PTTCommunicationManager *) doGetCommunicationManager;
```

//4、设置服务器的所有事件响应 delegate

```
- (void) setDelegate:(id<PTTConnectionManagerDelegate>)delegate;
```

//兼容 Swift, 功能同上

```
- (void) doSetDelegate:(id<PTTConnectionManagerDelegate>)delegate;
```

### 3.3.2 服务器回调 Callback

异步回调服务器的所有事件响应, 即从服务器收到的事件  
PTTConnectionManagerDelegate (PTTConnectionManager.h)

//1、用户已加入 (即变为可见了)

```
- (void) userAdded:(PTTUser *)user;
```

//2、用户已离开（即变为不可见了）

- (void) userLeft:(PTTUser \*)user;

//3、该帐号在其他设备登录或者被踢出服务器

- (void) imKicked;

//4、用户讲话状态发生改变

- (void) userTalkingStateChanged:(PTTUser \*)user;

//5、收到所加入的频道

- (void) channelAdded:(PTTChannel \*)channel;

//6、频道已删除

- (void) channelRemoved:(PTTChannel \*)channel;

//7、用户头像发生改变

- (void) userTextureChanged:(PTTUser \*)user;

//8、用户声音源发生改变(目前只表示是否用了手咪)

- (void) userAudioSourceChanged:(PTTUser \*)user;

//9、用户昵称发生改变

- (void) userNickChanged:(PTTUser \*)user;

//10、用户上线（登上服务器）

- (void) joinedServerAsUser:(PTTUser \*)user;

//11、登录返回结果

- (void) loginResult:(PTTRejectReason)reason;

//12、因为权限问题而被拒绝

- (void) permissionDeniedForReason:(NSString \*)reason;

//13、注册返回结果：return\_id>0 成功

- (void) registerResult:(int)return\_id;

//14、登录后修改密码

- (void) passwordUpdated:(NSString \*)pwd;

//15、忘记密码密码重新设置：suc 为真是成功

- (void) forgetPasswordResult:(BOOL)suc;

//16、申请或归还话筒返回结果：apply 为真是申请，suc 为真是成功

- (void) applyMicResult:(BOOL)apply success:(BOOL)suc;

//17、用户已被搜到

- (void) userSearched:(PTTUser \*)user;

//18、好友信息变化

- (void) contactChanged;

//19、收到好友申请

- (void) applyContactRecieved;

//20、被邀请进临时频道

- (void) belInvitedTotmpChannel:(PTTChannel \*)chan;

//21、全体通知更新

- (void) entUpdated;

//22、停止讲话

- (void) stopTalk;

//23、监听频道改变

- (void) listenStateChanged:(BOOL)listen;

//24、收到新消息

- (void) messageDidAdd:(PTTChatMessage \*)cm;

//25、更新消息

- (void) messageDidUpdate:(PTTChatMessage \*)cm;

### 3.3.3 服务器接口 API

负责管理与服务器的通信，即发送给服务器的事件  
PTTCommunicationManager (PTTCommunicationManager.h)

//1、用户自己

- (PTTUser \*) connectedUser;

//2、获取 session 值对应的用户

- (PTTUser \*) userWithSession:(NSUInteger)session;

//3、ID 索引频道信息

- (PTTChannel \*) channelWithId:(NSUInteger)channelId;

//4、频道用户表

- (NSMutableDictionary \*) channelMap;

//5、好友表

- (NSMutableDictionary \*) contactMap;

//6、待确认的好友表

- (NSMutableDictionary \*) pendingContactMap;

//7、新加入频道

- (void) joinChannel:(int)chanId pwd:(NSString \*)pwd;

//8、进入频道

- (void) enterChannel:(int)chanId;

//9、创建频道

- (void) createChannel:(NSString \*)channelName //创建普通频道才用，否则@""  
pwd:(NSString \*)channelPwd //创建普通频道才用，否则@""  
members:(NSString \*)channelMembers //创建临时频道时才用，否则@""  
searchable:(BOOL)searchable //为真是频道公开，可让人搜到  
tmp:(BOOL)tmp; //为真是创建临时频道，否则普通频道

//10、重命名频道

- (void) renameChannel:(int)channelId chanName:(NSString \*)channelName;

//11、修改频道密码

- (void) changeChannelPwd:(int)channelId chanPwd:(NSString \*)channelPwd;

//12、公开频道，让人可以搜到：pub 为真是允许公开

- (void) setChannelPub:(int)channelId searchable:(BOOL)pub;

//13、删除频道

- (void) deleteChannel:(int)channelId;

//14、退出频道

- (void) quitChannel:(int)channelId;

//15、申请或归还话筒：apply 为真是申请，否则是归还

- (void) applyMic:(int)channelId applyOrGiveBack:(BOOL)apply;

//16、上报通话状态：talk 为真是正在通话

- (void) reportTalkingState:(BOOL)talk;

//17、搜索用户

- (void) searchUser:(NSString \*)keyword;

//18、添加或删除好友：\_add 为真是添加，否则是删除

- (void) applyContact:(int)uid add:(BOOL)\_add;

//19、设置头像

- (void) setAvatar:(UIImage \*)image;

//20、设置昵称

- (void) setNick:(NSString \*)nick;

//21、设置密码

- (void) setPassword:(NSString \*)pwd;

//22、上报声音源(目前只表示是否用了手咪)：1 表示手咪，0 表示不是

- (void) reportAudioSource:(int)source;

//23、拒绝好友申请

- (void) deletePendingContact:(int) uid;

//24、设置权限

- (void) manageMember:(int)cid uid:(int)uid opt:(int)ban;

//25、组织信息

- (PTTEnt \*) currentEnt;

//26、组织信息

- (PTTUser \*) userWithId:(int)userId ;

//27、发送全体通知

- (void) setEntAnnonuce:(NSString \*)announce;

//28、发送频道通知

- (void) setChannelAnnonuce:(int)channelId announce:(NSString \*)announce;

//29、查询频道通知

- (NSString \*) getChannelAnnonuceByChannelId:(int) channelId;

//30、保存频道通知

- (void) saveChannelAnnonuceByChannelId:(int)channelId announce:(NSString \*)announce;

//31、设置开启定位

- (void) setLocation:(BOOL)on;

//32、频道开启定位的用户数

- (int)locationUsersInChannel:(NSUInteger)channelId;

//33、获取频道用户的位置的 url

- (NSString \*) getUrlForDownloadLocaton:(int)channelId recent:(BOOL)isRecent;

//34、监听频道信息

- (NSArray \*) listenChannels;

//35、设置监听频道

- (void) setListenChan:(int)cid listen:(BOOL)listen;

//36、频道是否监听

- (BOOL) isListen:(int)cid;

//37、清除未读消息

- (void)clearUnreadMessageCountByChannelId:(int) channelId;

//38、获取频道的所有消息

- (NSMutableArray \*) getChatMessagesByChanId:(int) channelId;

//39、文件服务器地址

- (NSString \*) fileHost;

//40、媒体文件的本地路径

- (NSString \*) allPathWithServerPath:(NSString \*)serverPath;

//41、生成媒体文件的服务器保存路径

- (NSString \*) generateServerPath:(NSString \*)ext;

//42、保存媒体数据

- (void)saveData:(NSData \*)data toServerPath:(NSString \*)serverPath;

//43、保存媒体文件

- (void)saveFile:(NSString \*)filePath toServerPath:(NSString \*)serverPath;

//44、媒体文件是否存在

- (BOOL) fileExistsAtServerPath:(NSString \*)serverPath;

//45、头像的本地路径

- (NSString \*) avatarPathWithServerPath:(NSString \*)serverPath;

//46、生成头像的服务器保存路径

- (NSString \*) generateAvatarServerPath:(NSString \*)ext;

//47、保存头像数据

- (void)saveAvatarData:(NSData \*)data toServerPath:(NSString \*)serverPath;

//48、发送文字消息

- (void)sendTextMessage:(NSString \*)text channelId:(int)channelId;

//49、发送图片消息

- (void) sendPictureMessage:(NSData \*)thumbnail text:(NSString \*)text size:(int)size  
channelId:(int)channelId;

//50、发送视频消息

- (void) sendVideoMessage:(NSData \*)thumbnail text:(NSString \*)text size:(int)size duration:(int)duration  
channelId:(int)channelId;

//51、发送定位消息

- (void) sendLocationMessage:(NSData \*)thumbnail text:(NSString \*)text channelId:(int)channelId;

//52、实时通知

- (void) pushNotice:(NSString \*)notice //notice:通知内容  
targetType:(int)targetType //targetType:0-面向组织全体人员；1-频道所有在线成员；2-某个人  
3-在指定频道里的人员  
targetId:(int)targetId //targetId:type=0 时忽略；1 和 3 时频道 id，2 时用户 id  
senderType:(int)senderType //senderType:1-以调度员身份发送；2-以频道管理员身份发送  
ttsPlay:(int)ttsPlay; //ttsPlay:1-语音播报；0-仅文字

### 3.3.4 声音接口 API

声音操作 PTTAudio (PTTAudio.h)

//1、启动

- (BOOL) start;

//1、开始录音：start 为真是开始，needOpenSco 为真是蓝牙

- (void) startRecording:(BOOL)start needOpenSco:(BOOL)needOpenSco;

//2、正在录音

- (BOOL) isRecording;

//3、app 退出时，调用

-(void) quit;

//5、回放 ptt

- (void) playbackPtt:(NSData \*) data url:(id)url delegate:(id)delegate;

//6、停止回放 ptt

- (void) stopPtt;

### 3.3.5 声音回调 Callback

声音回调 PTTPlaybackVoiceDelegate (PTTAudio.h)

//1、回放开始

- (void)voiceDidStart:(id)url;

//2、回放结束

- (void)voiceDidStop:(id)url;

### 3.3.6 蓝牙手咪接口 API

管理蓝牙手咪操作 PTTBtServer (PTTBtServer.h)

//1、初始化

-(void)initBLE;

//2、获取手咪状态

-(HandmicState)getTargetHandmicState;

//3、获取手咪型号

-(NSString \*)getHandmicModel;

//4、获取手咪固件

-(NSString \*)getHandmicFirmware;

//5、初始化是否完成

-(BOOL)btInited;

//6、开始扫描

-(void)startScan;

//7、停止扫描

-(void)stopScan:(BOOL)withOutEvent;

//8、连接

-(void)connect:(PTTPeriperallInfo \*)peripherallInfo withFinishCB:(eventBlock)callback;

//9、断开

```
-(void)disconnect;
```

```
//10、发现服务
```

```
-(void)discoverService:(CBService*)service;
```

```
//11、读取 character
```

```
-(void)readValue:(CBCharacteristic*)characteristic;
```

```
//12、获取验证值
```

```
-(void)readAuth;
```

### 3.3.7 蓝牙手咪回调 Callback

蓝牙手咪回调 PTTBtServerDelegate (PTTBtServer.h)

```
//1、停止扫描完成
```

```
-(void)didStopScan;
```

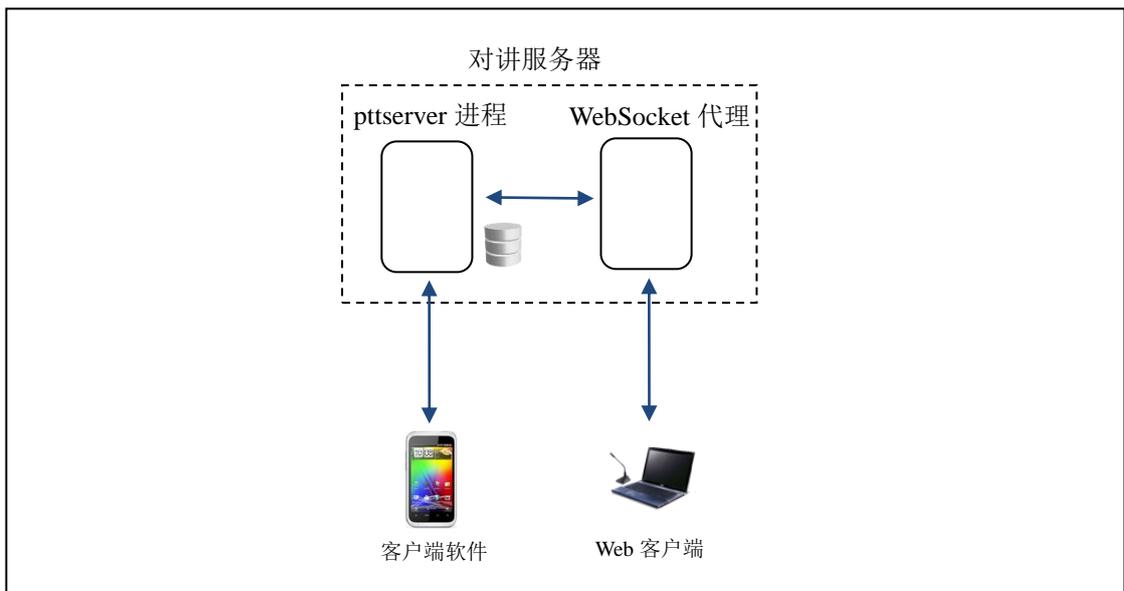
```
//2、已经发现的外设发生，例如外设名称
```

```
-(void)discoveredPeripheralChanged;
```

## 4 web 版集成

### 4.1 概述

滔滔支持浏览器无插件对讲客户端。Html5 版本对讲使用 html、css、JavaScript 等语言开发，实现不依赖插件，直接网页中对讲。平台架构：



Web 客户端软件基于 HTML5 技术实现。滔滔平台使用 TCP 传输控制指令（登录、数据同步等），使用 UDP 传输语音数据（可选 TCP）。

在浏览器中，无法直接使用 TCP 和 UDP 通道，须使用 WebSocket 方式进行通信。为此，在服务器端需运行一个 websocket 代理程序，负责把浏览器发来的 websocket 数据转发到主服务器相应端口，下行数据同样需要代理程序中转。

同时，Web SDK 内部实现语音的实时编解码。

SDK 可运行于所有支持 HTML5 的浏览器中。

## 4.2 集成步骤

### 4.2.1 服务器部署

首先确保对讲服务器已正常运行（参考《对讲服务器部署指南(docker 方式)》）。注意，Docker 镜像里已集成 websockify，无需额外操作。如果需要在其他服务器上运行 WebSocket 代理程序，则需以下操作。

WebSocket 代理程序可运行在任意服务器上，可以跟 pttserver 同机，也可以不同机器、不同区域。在一般场合下，不妨与 pttserver 程序同机运行。

1) 在服务器上，安装 websockify 程序：

```
sudo apt-get install websockify
```

2) 为 https 协议准备 cert 和 key 两个证书文件：

测试时，可运行以下命令，并根据实际情况填写信息，生成 mycert.crt 和 mykey.key 文件：

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048  
-keyout ./mykey.key -out mycert.crt
```

使用自行生成的证书时，浏览器打开链接时，会提示风险。正式使用时，建议使用 ZeroSSL 等工具获取正式的证书。

3) 准备好我们提供的 dist 目录，并运行 websockify：

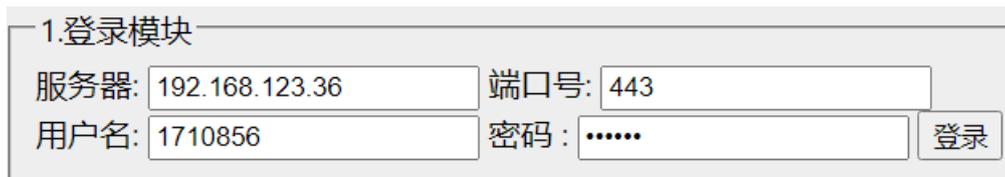
```
sudo websockify --cert=mycert.crt --key=mykey.key --ssl-only  
--ssl-target --web=./dist 443 totalkd.allptt.com:59638
```

其中，mycert.crt、mykey.key、./dist 均替换成实际路径 totalkd.allptt.com 应替换成自己服务器的网络地址。

### 4.2.2 浏览器访问验证

在 PC 上，暂时关闭防火墙，并打开浏览器（推荐 chrome）

chrome 浏览器 https 方式访问 websockify 所在机器地址，如 https://192.168.123.120



1.登录模块

服务器:	<input type="text" value="192.168.123.36"/>	端口号:	<input type="text" value="443"/>
用户名:	<input type="text" value="1710856"/>	密码:	<input type="password" value="....."/>

服务器栏填写 websockify 所在机器地址，如 192.168.123.120，端口号 443，用户名密码填写 pttserver 上已创建的账号，点击“登录”。

登录成功后，会显示登录状态、当前频道、频道和用户列表等信息。

## 4.3 接口说明

### 4.3.1 登录

//1、登录 API

```
window.totalkAPI.userLogin(ipaddress, port, username, pwd);
```

//2、退出登录 API

```
window.totalkAPI.userLogout();
```

//3、登录成功的回调 callback

```
window.totalkAPI.setServerSyncCallback(() => {  
});
```

//4、登录失败的回调 callback

```
window.totalkAPI.setUserLoginCallback((type, reason) => {  
});
```

### 4.3.2 频道

//1、获取所有频道信息 API

```
window.totalkAPI.getAllChannels();
```

//2、切换频道 API

```
window.totalkAPI.userEnterChannel(channelId);
```

//3、设置切换频道的回调 callback

```
window.totalkAPI.setEnterChannelCallback((channelId) => {  
});
```

//4、监听频道 API

```
window.totalkAPI.userListenChannel(channelId);
```

//5、取消监听频道 API

```
window.totalkAPI.userUnlistenChannel(channelId);
```

//6、设置监听频道的回调 callback

```
window.totalkAPI.setListenStateCallback((channelId, listen) => {  
});
```

//7、新增频道的回调 callback

```
window.totalkAPI.setNewChannelCallback((channel) => {
});

//8、设置删除频道的回调 callback
window.totalkAPI.setRemoveChannelCallback(() => {
});

//9、设置更新频道的回调 callback
window.totalkAPI.setUpdateChannelCallback(() => {
});

//10、创建频道
var payload = {
  channel_name: gMiniProgramChannelName,
  channel_pwd: gMiniProgramChannelPwd,
  temporary:gMiniProgramChannelTemporary, //bool, true: 为选呼, 即创建临时频道
  members:"" //format: "1001001,1001002,1001055", etc, include creator
}
window.totalkAPI.userCreateChannel(payload);
```

### 4.3.3 用户

```
//1、获取所有用户信息 API
window.totalkAPI.getAllUsers();

//2、设置新增用户的回调 callback
window.totalkAPI.setNewUserCallback(() => {
});

//3、设置删除用户的回调 callback
window.totalkAPI.setRemoveUserCallback(() => {
});

//4、设置更新用户的回调 callback
window.totalkAPI.setUpdateUserCallback(() => {
});
```

### 4.3.4 对讲

```
//1、鼠标点击按下 PTT 键起麦 API
window.totalkAPI.userPttDown();

//2、鼠标点击松开 PTT 键放麦 API
```

```
window.totalkAPI.userPttUp();
```

//3、设置 PTT 抢麦的回调 callback

```
window.totalkAPI.setApplyMicCallback((apply, success) => {  
});
```

//4、设置 PTT 通话的回调 callback

```
window.totalkAPI.setUserTalkingCallback((userId, talking) => {  
});
```

//5、设置 PCM 的回调 callback, 每次通话完毕后一段 PCM 语音

```
window.totalkAPI.setPcmFrameCallback((pcmAudio) => {  
});
```

### 4.3.5 收发即时通信消息

//1、发送消息

```
window.totalkAPI.userSendInstantMessage(payload);
```

调用时, 先构造 payload:

```
var payload = {  
  from_id: 1735773,  
  target_id: 191083,  
  message_type: 0,  
  content: "hello"  
}  
window.totalkAPI.userSendInstantMessage(payload);
```

其中 payload 里可选参数列表:

```
from_id    //发送者用户号  
target_id  //所在频道 id  
message_type //消息类型 0-文字 1-语音 2-图片 3-短视频 4-直播 5-位置 6-其他文件, 7-抢单  
content    //文本内容。文字信息内容, 或媒体、文件的 url  
time       //时间戳  
thumbnail  //缩略图  
size       //文件大小, 单位字节数, 仅对文件消息有效  
duration   //媒体文件时长, 仅对语音、短视频、直播视频有效  
orig_name  //文件原始名字, 仅对文件消息有效  
chan_seq_id //频道内消息流水号  
from_nick  //发送者昵称
```

说明: 发送 5-位置时, 可设置 content (必选) 和 thumbnail (可选)。其中 thumbnail 建议 5KB 以下, content 格式形如: "<经度>----<纬度>----<地点名称>----<详细地址>"。例如:

"116.36----38.33----玉渊潭----北京市西城区月坛南街 33 号"。如果没有地点名称和详细地址, 相应位置应留空格: "116.36----38.33---- ----"。

## //2、收到即时通信消息回调

```
window.totalkAPI.setInstantMessageCallback((from_id, target_id, message_type, content, time,
thumbnail, size, duration, orig_name, chan_seq_id, from_nick) => {

    });
```

## //3、发送图片、短视频、文件

发送媒体文件分为两步, 首先把文件上传到服务器, 获得一个 url, 然后调用 `userSendInstantMessage`, 并携带把 url 和其他必要信息。其他人收到消息, 提取 url, 并下载原始媒体文件。以发送本地 `mypic.jpg` 为例, 具体步骤:

### 1) 确定服务器保存文件的相对路径

服务器路径并非强制, 但为了便于管理并与其他客户端保持一致, 建议规则是:

```
media/<组织 ID>/<yyyyMMdd 格式日期>/<自己用户 ID/10000>/<自己用户 ID>/<16 位随机串>.<
文件原扩展名>
```

例如:

```
media/10000/20230411/100/1001001/adffsdfewdfsfgsb.jpg
```

### 2) 上传文件

上传时, 应指定上传路径, 规则是:

```
http://<服务器 IP:端口>/TotalkWar/uploadFile.do?filename=<第 1)步生成的服务器路径>, 例如:
```

```
http://192.168.1.100:59680/TotalkWar/uploadFile.do?filename=
```

```
media/10000/20230411/100/1001001/adffsdfewdfsfgsb.jpg
```

### 3) 上传成功, 发送消息

确认第 2)步文件上传成功后, 发送消息:

```
var payload = {
    from_id: 1001001,
    target_id: 1001,
    message_type: 2,
    content: " media/10000/20230411/100/1001001/adffsdfewdfsfgsb.jpg",
    size:300000,
    thumbnail:<二进制缩略图, 建议 5KB 内>
}
window.totalkAPI.userSendInstantMessage(payload);
```

## 4.3.6 接收视频直播

### //1、设置直播的回调 callback, 直播开始和结束时

```
window.totalkAPI.setUserCastingCallback((
    userId,    //用户号
    nick,      //用户昵称
    channelId, //频道号
    start,     //true-开始, false-结束
    castUrl,   //直播地址
```

```
videoUrl, //视频回放地址（结束的时候）
time, //时间
duration, //时长（毫秒）
thumbnail //缩略图
) => {
});
```

从 2023.11 月开始，直播支持低延迟播放。播放方式：首先，castUrl 这个 rtmp://形式的 url 要转化为低延迟 url，转化规则：

castUrl: rtmp://<server IP>:1935/live/11223344

低延迟 url: web rtc:// <server IP>/live/11223344

javascript 播放可参考：

<https://github.com/kernelj/jswebrtc>

在访问 web rtc://链接时，会收到服务器返回的 api 接口地址，默认是 http://形式。如果客户的对讲页面是 https 方式的，可能会提示：'http://test.allptt.com:1985/rtc/v1/play/'. This request has been blocked; the content must be served over HTTPS.

解决方法：修改 jswebrtc 代码，把 <http://test.allptt.com:1985> 转换为 <https://test.allptt.com:1990>，即可。

修改要点：

- 1) 全局搜 1985，改为 1990；
- 2) 在 src/player.js 的 Player.prototype.startLoading = function () { 里找到  
var url = 'http://' + \_self.urlParams.server + ':' + port + api;  
把 http://改为 https://

注意：如果使用 https 接口，在 docker 部署时，需要按 docker 文档，正确准备服务器证书。

### 4.3.7 推送文字播报

说明：本功能可推送一段文字，到组织全体人员或指定频道、用户。接收端会弹窗显示文字内容，并播报语音。

//1、推送文字

```
var payload = {
  senderType:2, //0:平台(web 后台组织管理员); 1:调度员; 2:频道管理员;
  targetType: 1, //0 组织全体人员 1 指定频道所有在线成员 2 指定用户 3 指定频道、在频道里
  的人员
  targetId: 256392, //与 targetType 一起生效。 type 为 0 时，id 忽略 type 为 1 和 3 时，id 表
  示频道 id type 为 2 时，id 表示用户 id
  content: "PushNotify Test", //文字内容，1000 中文、 2000 字符内
  ttsPlay: 1, //是否播报语音 0 为否，1 为是。如果是 0 则终端只显示文字，不播报语音，如果
  是 1 则终端既显示文字，又播报语音 。
}
```

```
window.totalkAPI.userPushNotifyMessage(payload);
```

### 4.3.8 收发公告

说明：本接口可发送公告到组织全体人员或频道成员。客户端收到后可滚动显示。一般用于不紧急的通知等。

```
//1、发送组织公告（仅组织调度员可操作）
    var payload = {
        ent_announce:"组织公告测试！", //组织公告只能组织调度员发送
    }
    window.totalkAPI.userSendEntAnnounceMessage(payload);
//2、接收组织公告
    window.totalkAPI.setEntAnnounceCallback((payload) => {
        console.log("setEntAnnounceCallback", payload)
    });
//3、发送频道公告（仅频道管理员可操作）
    var payload = {
        channel_id:1001, // 频道公告只能频道管理员发送
        announce:"频道公告测试",
    }
    window.totalkAPI.userSendChanAnnounceMessage(payload);
//4、接收频道公告
    window.totalkAPI.setChanAnnounceCallback((payload) => {
        console.log("setChanAnnounceCallback", payload)
    });
```

### 4.3.9 查询离线消息（历史消息查询）

说明：本接口可查询指定渠道的历史消息。

//1、历史消息查询接口

```
needMoreRecords(int channelId, long fromSequence, int pageSize);
```

//channelId:频道 ID;

//fromSequence: 从哪条消息开始，由新到旧查询。如果不能确定哪条开始，就传入-1，则从本频道最新一条开始。

//pageSize: 返回几条记录。

查询结果，会通过以下回调返回：

```
// 历史消息查询的回调 callback
window.totalkAPI.onOfflineMessagesLoaded((payload) => {
    console.log("onOfflineMessagesLoaded", payload)
});
```

### 4.3.10 发起音视频通话

本功能体验类似微信的多人音视频通话。

本节及《应答音视频通话》部分，只实现了通话消息通知。具体音视频处理，需要第三方自己实现

可参考开源实现，如 <https://github.com/versatica/mediasoup-demo> 或  
<https://gitee.com/DENNIS881219/mediasoup-demo> 此项目有在线演示：  
<https://v3demo.mediasoup.org>

//1、发起音视频通话接口

```
var payload = {
    //target_type 为 0 时，members 是用户 id 列表，形如"1001001,1001002"。
    // 1 忽略
    // 为 2 时，members 是个组合，形如 "dept:xxx;chan:xxx;user:xxx"，气质 xxx 也是 id 的组
    合，逗号分割
    target_type:0,
    members:"1001001,1001002",
    room_id:"aabbccdd" //8 位随机字符串
}
window.totalkAPI.userStartMeeting(payload);
```

收到别人发起的通话：

```
window.totalkAPI.setStartMeetingCallback((from_id, from_nick, room_id) => {
    console.log('startMeeting');
    console.log(from_id); //呼叫者用户号
    console.log(from_nick); //呼叫者昵称
    console.log(room_id); //meeting room id
});
```

### 4.3.11 应答音视频通话

//1、应答音视频通话接口

```
var payload = {
    meeting_id:"aabbccdd", //应答对应的 room id
    callee_state:2, //1:主呼； 2： 接受； 3： 拒绝； 4： 占线； 5： 主叫自行取消 6:对方无应答
    to_ids:"1001001" //发给哪些人。对于别人发起的通话，只发给发起者即可
}
window.totalkAPI.userAnswerMeeting(payload);
```

```
function testAnswerMeeting() {
    var payload = {
        meeting_id:"aabbccdd", //应答对应的 room id
        callee_state:2, //1:主呼； 2： 接受； 3： 拒绝； 4： 占线； 5： 主叫自行取消 6:对方无应答
        to_ids:"1001001" //发给哪些人。对于别人发起的通话，只发给发起者即可
    }
    window.totalkAPI.userAnswerMeeting(payload);
}
```

自己发起的通话，收到别人的应答：

```
// 设置 收到音视频呼叫应答的 callback
window.totalkAPI.setAnswerMeetingCallback((from_id, meeting_id, callee_state) => {
    console.log('answerMeeting');
    console.log(from_id);    //应答者用户号
    console.log(meeting_id); //meeting room id
    console.log(callee_state); //应答者操作。//1:主呼； 2: 接受； 3: 拒绝； 4: 占线； 5: 主叫自行取消 6:对方无应答
});
```

## 5 后台管理接口

### 5.1 概述

服务器采用 C++/Qt 开发。

目前的客户端 SDK，是针对用户自注册的场景。对于企业用户，第三方系统可能无需支持自注册。此时，用户、频道、成员都可通过后台管理。

服务器启动后，会提供 Web Service 服务。第三方可自行开发 Web 或原生管理界面，通过 http 协议对服务器数据进行增删改查操作。目前提供基于 VUE 框架的前端代码参考。

注意：

- 1) 所有接口调用都采用 POST 方法，且 Content-Type 必须设为 application/x-www-form-urlencoded。所有接口均可用 postman 软件测试验证。请求方式参考下图：

The screenshot shows a Postman interface for a POST request to `http://192.168.1.100:8081/admin/?method=login`. The request body is `x-www-form-urlencoded` with the following data:

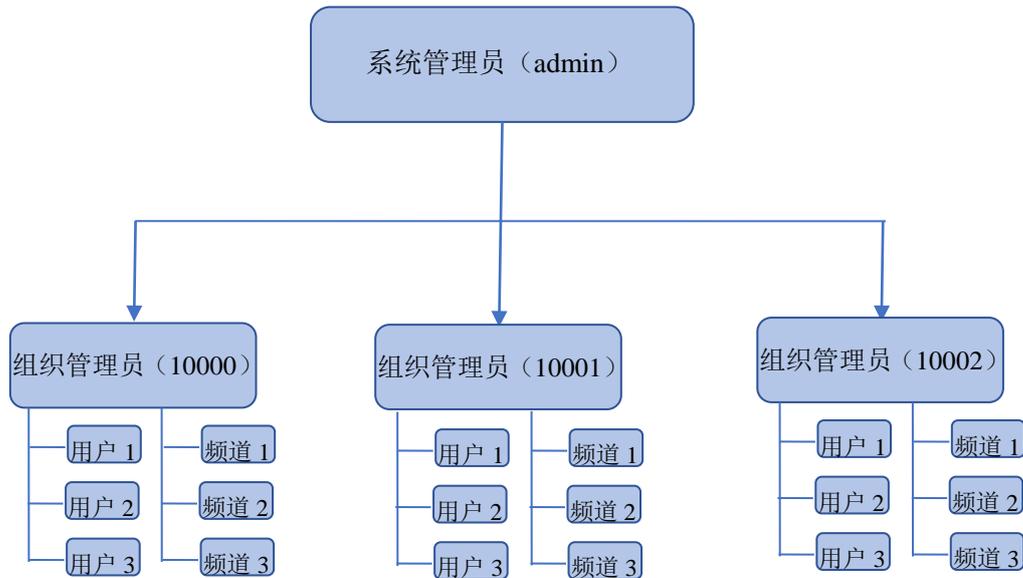
KEY	VALUE	DESCRIPTION
account	admin	
password	123456	

The response is a JSON object:

```
1 {
2   "code": 200,
3   "reason": "success",
4   "token": "eyJhbGciOiAiSFMyNTYiLCJ0eXAiOiAiS1dUIH09LmEyJhdWQiOiAiYWRTaW4iLCJleHAiOiAiMTY2MjY5MDU4MyJ9.
5     ABh0YcCx1KGYaRD2CVMXLKJdHD90YgQt24BBs0IFQz0="
```

- 2) 接口设计是提供后台管理，最多响应 1000 个连接，所以 app 不要直接访问。

#### 5.1.1 后台业务规则



服务卡分为永久卡和月卡两种。

首先，系统管理员 (admin) 登录并购买服务卡；然后，系统管理员把服务卡输入系统；最后，系统管理员新建组织并为每个组织分配服务卡数量，只有已分配了服务卡数量的组织，组织管理员才可以新建用户。

组织管理员 (账号组织 ID) 登录并创建用户时，需指定卡类型，将消耗本组织的卡。永久卡可以供一名用户永久使用，一张月卡供一个用户使用一个月。删除用户时，用户未使用的服务卡将被回收，重新回到组织的卡池。

## (1) 系统管理员身份登录

账号为 admin，密码默认为 123456。登录 (接口 5.2.1.1 login) 成功后，返回 token，后续所有系统管理员的接口都需要带上此 token。IP 地址填写后台服务器地址。



手机对讲系统

用户名: admin

密码: .....

IP地址: 192.168.50.11:8080

登录 重置

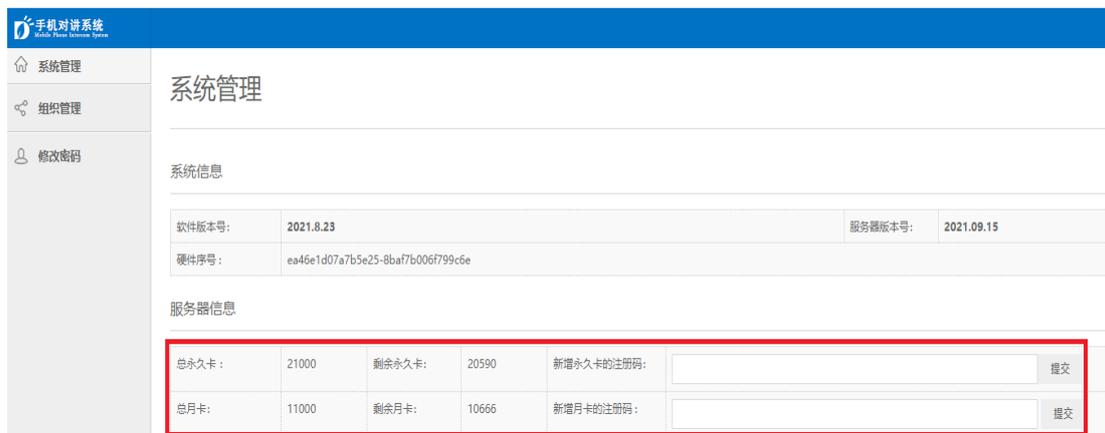
## (2) 购买服务卡

系统管理员获取系统配置（接口 5.2.1.3 getSystemConfig）中的硬件序号（hardware\_id），使用此硬件序号向我司购买服务卡，购买成功后将分别得到服务卡的 40 位字符串注册码。



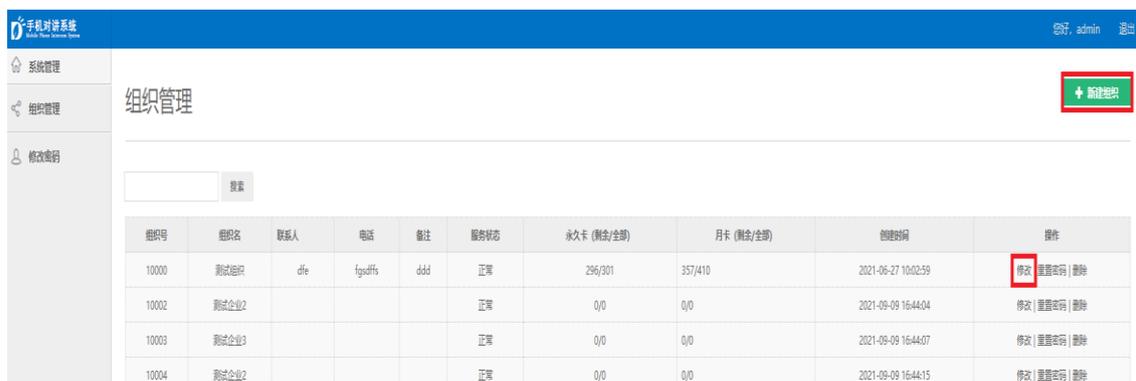
### (3) 服务卡输入系统

系统管理员修改系统配置（接口 5.2.1.4 setSystemConfig）中的增加永久卡（add\_perm）或增加月卡（add\_month），填入购买到的相应的 40 位字符串注册码，提交成功后，可以获取系统配置（getSystemConfig）的总永久卡（all\_perm）、剩余永久卡（remain\_perm）、总月卡（all\_month）、剩余月卡（remain\_month），总卡数量是购买得到的，剩余卡数量是仍可以用来分配给组织的。



### (4) 新建组织并分配服务卡数量

系统管理员新建组织（接口 5.2.2.1 createEnt），得到组织 ID（比如 10000）。



系统管理员修改组织属性（接口 5.2.2.8 setEntProperty）为组织分配永久卡数量（add\_perm）或月卡数量（add\_month）。只有已分配服务卡数量的组织，其组织才能新建用户。



手机对讲系统

系统管理 | 组织管理 | 修改密码

### 修改组织

组织属性	
名称	
组织ID:	10000
组织名称:	测试组织
创建时间:	2021-06-27 10:02:59
联系人:	dfe
电话:	fgsdffs
备注:	ddd
是否暂停服务:	0
已购永久卡:	410
剩余永久卡:	357
新增永久卡数量:	<input type="text" value="0"/> <input type="button" value="提交"/>
已购月卡:	301
剩余月卡:	296
新增月卡数量:	<input type="text" value="0"/> <input type="button" value="提交"/>

## (5) 组织管理员身份登录

账号为已新建的组织 ID 比如 10000，密码默认为 123456。登录（接口 5.3.1.1 login）成功后，返回 token，后续所有组织管理员的接口都需要带上此 token。IP 地址填写后台服务器地址。

- 注意：1) 不要每次调用都获取一次 token，而应该拿到 token 后重复使用。  
 2) token 里如果有“+”，不能直接使用，需将+替换为%2B。



手机对讲系统

用户名: 10000

密码: .....

IP地址: 192.168.50.11:8080

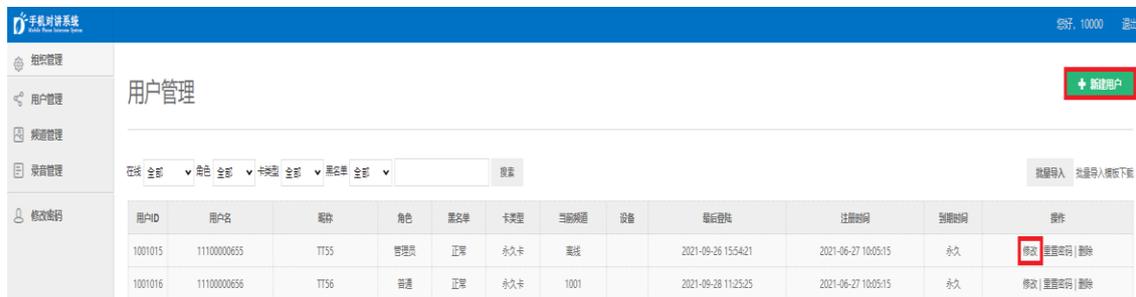
## (6) 新建用户并分配服务卡类型

组织管理员获取本组织属性（接口 5.3.1.3 getEntProperty）已购永久卡（all\_perm）、剩余永久卡（remain\_perm）、已购月卡（all\_month）、剩余月卡（remain\_month）。已购卡数是由系统管理员分配得到，只有剩余卡数不为 0 才能新建用户。



组织信息	用户信息	频道信息
组织名称		
组织名称	测试组织	<input type="text"/> 修改
创建时间	2021-06-27 10:02:59	
是否暂停服务	否	
已购永久卡	410	
剩余永久卡	357	
已购月卡	301	
剩余月卡	296	
当前服务器数据存储天数 (包括录音、聊天文件、上传位置):	7	
设置服务器数据存储天数 (0表示不录音):	<input type="text" value="0"/>	提交

组织管理员可以新建用户或修改用户。



用户ID	用户名	昵称	角色	黑名单	卡类型	当前频道	设备	最后在线	注册时间	到期时间	操作
1001015	11100000655	TT55	管理员	正常	永久卡	离线		2021-09-26 15:54:21	2021-06-27 10:05:15	永久	<input type="button" value="修改"/> <input type="button" value="重置密码"/> <input type="button" value="删除"/>
1001016	11100000656	TT56	普通	正常	永久卡	1001		2021-09-28 11:25:25	2021-06-27 10:05:15	永久	<input type="button" value="修改"/> <input type="button" value="重置密码"/> <input type="button" value="删除"/>

组织管理员新建用户（接口 5.3.2.1 createUser），为新用户分配服务卡类型，永久卡（perm）或月卡（month），默认为永久卡（perm）。每新建一个用户，该组织将消耗一张卡，相应的剩余卡数量自动减 1。

手机对讲系统

组织管理  
 用户管理  
 频道管理  
 录音管理  
 修改密码

## 新建用户

用户信息

<b>名称</b>		
用户名(禁止7位数字):	<input type="text"/>	用户默认密码为123456
卡类型:	<input checked="" type="radio"/> 永久卡 <input type="radio"/> 月卡	
昵称:	<input type="text"/>	默认使用用户名
	<input type="button" value="确定"/>	

组织管理员修改用户（接口 5.3.2.6 setUserProperty），可以更换服务卡类型，可以为用户增加月卡数量。

手机对讲系统

组织管理  
 用户管理  
 频道管理  
 录音管理  
 修改密码

## 修改用户

用户属性

名称	内容	
用户昵称:	<input type="text" value="TT56"/>	
剩余永久卡/ 剩余月卡:	<input type="text" value="/ 296"/>	月卡数量: <input type="text"/>
卡类型:	<input checked="" type="radio"/> 永久卡 <input type="radio"/> 月卡	
角色:	<input checked="" type="radio"/> 普通 <input type="radio"/> 调度员	
黑名单:	<input checked="" type="radio"/> 正常 <input type="radio"/> 黑名单	

## (7) 新建频道并添加成员

组织管理员新建频道（接口 5.3.3.1 createChannel），可以查看频道，把已新建的用户添加到相应的频道（5.3.4.1 addMember）。

手机对讲系统 登录, 10000 退出

组织管理  
 用户管理  
 频道管理  
 录音管理  
 修改密码

## 频道管理

编号	频道编号	频道名称	频道成员	操作
1	1001	2332	<input type="text" value="20 套组"/>	编辑   删除
2	1002	频道2	<input type="text" value="8 套组"/>	编辑   删除

编号	用户名称	昵称	成员身份	成员群组	操作
1	1001015	TT55	普通成员	普通	删除
2	1001016	TT56	普通成员	普通	删除

## 5.1.2 返回码 (code)

返回码 (code)	原因描述 (reason)
<b>200</b>	<b>success</b> (成功)
<b>401</b>	<b>url error: only support POST</b> (url 错误: 只支持 POST 方法) <b>url error: missed method</b> (url 错误: url 中缺失 method) <b>url error: path</b> (url 错误: 路径错误, 应该是/admin/或者/ent/) <b>url error: api name</b> (url 错误: api 方法名错误)
<b>402</b>	<b>parameter missed: xxx</b> (缺失参数: xxx)
<b>403</b>	<b>parameter error: xxx</b> (错误参数: xxx, 比如 eid 或 uid 不存在、输入不合法等)
<b>404</b>	<b>login failed</b> (登录失败, 请确认账号密码是否正确)
<b>405</b>	<b>token invalid</b> (token 无效)
<b>406</b>	<b>token timeout</b> (token 超时)
<b>407</b>	<b>token error: api</b> (token 错误: api 方法不属于 token 身份) <b>token error: eid</b> (token 错误: token 可操作的 eid 不存在, 比如被删了)
<b>500</b>	<b>operation error</b> (执行错误)

## 5.2 系统管理员接口说明 (/admin/)

### 5.2.1 系统管理

账号 account 为: admin

默认密码 password 为: 123456

前端调用 login 接口登录成功后获取到 token, 并保留此 token。其它所有接口都

必须带上这个 token 作为参数，用来验证身份和超时（1 小时）。

### 5.2.1.1 登录 (login)

#### ●parameters

参数名	参数类型	说明
account	string	账号，必填(admin)
encrypt	string	是否加密（0-不加密，用明文；1-加密），默认 0
password	string	密码，必填，6 位以上明文密码或其加密结果，默认不加密、123456。 如果加密，则需先调用 5.4.2 获取加密数据的公钥 <code>getPublicKey</code> ，然后用得到的 <code>publicKey</code> 对密码明文进行 RSA 加密

#### ●Request

POST <http://120.77.254.227/admin/?method=login>

```
{
  "account": "admin",
  "encrypt": "1",
  "password": ""
}
```

#### ●Response

```
{
  "code": 200, "reason": "success",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072 "
}
```

### 5.2.1.2 修改账号密码 (setAccountPassword)

#### ●parameters

参数名	参数类型	说明
encrypt	string	选填，是否加密（0-不加密，用明文；1-加密），默认 0

password	string	密码，必填，默认不加密、123456。如果加密则先调用 5.4.2 获取加密数据的公钥 <code>getPublicKey</code> ，然后用得到的 <code>publicKey</code> 对密码明文进行 RSA 加密
token	string	必填，登录获取的 token

### ●Request

POST <http://120.77.254.227/admin/?method=setAccountPassword>

```
{
  "encrypt": "1",
  "password": "",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ●Response

```
{
  "code": 200, "reason": "success"
}
```

## 5.2.1.3 获取系统配置 (getSystemConfig)

### ●parameters

参数名	参数类型	说明
key	string	服务器配置 key，必填
token	string	登录获取的 token，必填

### key 取值范围:

key (参数)	value (返回值)	说明
"version"	"2021.08.07"	服务器版本号
"hardware_id"	"62dee2f77f494846-207dc0fbfa260b93"	硬件序号
"file_server"	"192.168.123.34:8080"	文件服务器地址

"record_server"	"192.168.123.34:8080"	录音服务器地址
"meeting_server"	"192.168.123.34:8080"	通话服务器地址
"streaming_server"	"192.168.123.34:8080"	媒体服务器地址
"main_server_port_tcp"	"59638"	主服务器 TCP 端口
"main_server_port_udp"	"59638"	主服务器 UDP 端口
"all_perm"	"0"	总永久卡数量
"remain_perm"	"0"	剩余永久卡数量
"all_month"	"0"	总月卡数量
"remain_month"	"0"	剩余月卡数量

### ● Request

```
POST http://120.77.254.227/admin/?method=getSystemConfig
{
    "key": "version",
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNjc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ● Response

```
{
    "code": 200, "reason": "success",
    "value": "2021.08.07",
}
```

## 5.2.1.4 修改系统配置 (setSystemConfig)

### ● parameters

参数名	参数类型	说明
key	string	服务器配置 key, 必填
value	string	服务器配置 value, 必填

token	string	登录获取的 token, 必填
-------	--------	-----------------

**key 取值范围:**

key (参数)	value (参数)	说明
"file_server"	192.168.123.34:8080	设置文件服务器地址 (ip:端口号)
"record_server"	192.168.123.34:8080	设置录音服务器地址 (ip:端口号)
"meeting_server"	192.168.123.34:8080	设置通话服务器地址 (ip:端口号)
"streaming_server"	192.168.123.34:8080	设置媒体服务器地址 (ip:端口号)
"main_server_port_tcp"	59638	设置主服务器 TCP 端口 (1024-65534, 重启生效)
"main_server_port_udp"	59638	设置主服务器 UDP 端口 (1024-65534, 重启生效)
"add_perm"	40 位字符串的注册码	填写增加永久卡的注册码
"add_month"	40 位字符串的注册码	填写增加月卡的注册码

**● Request**

POST <http://120.77.254.227/admin/?method=setSystemConfig>

```
{
  "key": "streaming_server ",
  "value": "192.168.123.34:8080",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDC3OTkwMzkzLCJlc2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**● Response**

```
{
  "code": 200, "reason": "success"
}
```

## 5.2.2 组织管理 (创建、删除、获取、修改密码)

### 5.2.2.1 新建组织 (createEnt)

#### ●parameters

参数名	参数类型	说明
name	string	组织名, 必填
token	string	登录获取的 token, 必填

#### ●Request

POST <http://120.77.254.227/admin/?method=createEnt>

```
{
  "name": "testEnt",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200, "reason": "success",
  "eid": "10001", //新建组织的 ID (即 eid), 默认密码为 123456
}
```

### 5.2.2.2 删除组织 (deleteEnt)

#### ●parameters

参数名	参数类型	说明
eid	string	组织 ID, 必填
token	string	登录获取的 token, 必填

#### ●Request

POST <http://120.77.254.227/admin/?method=deleteEnt>

```
{
  "eid": "10001",
}
```

```

        "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
    }
    
```

● **Response**

```

    {
        "code": 200, "reason": "success"
    }
    
```

### 5.2.2.3 获取组织总数 (getEntsTotal)

● **parameters**

参数名	参数类型	说明
token	string	登录获取的 token, 必填

● **Request**

```

POST http://120.77.254.227/admin/?method=getEntsTotal
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

● **Response**

```

    {
        "code": 200, "reason": "success",
        "data": 2,
    }
    
```

### 5.2.2.4 分页获取组织列表 (getEntsInPage)

● **parameters**

参数名	参数类型	说明
pageIndex	int	页面索引，必填，从 0 开始
pageSize	int	页面组织数，必填
keyword	string	搜索关键词（包含搜索组织 ID 和组织名），必填
token	string	登录获取的 token，必填

### ●Request

POST <http://120.77.254.227/admin/?method=getEntsInPage>

```
{
  "pageIndex":0,
  "pageSize":20,
  "keyword":"","
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE5MzQ3OTkwMzZlc2VybWVtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
```

### ●Response

```
{
  "code": 200, "reason": "success",
  "total": 1, //搜索结果总数，而不是分页结果数量
  "data": [{"allMonth": 0,"allPerm": 0,"comment": "", "contact": "张三","createTime": "2021-06-27 10:02:59","eid": 10000,"name": "测试组织 1","pause": 0,"phone": "", "remainMonth": 0,"remainPerm": 0}], //pause:是否服务暂停（0为正常，1为暂停）
}
```

## 5.2.2.5 重置组织密码（resetEntPassword）

### ●parameters

参数名	参数类型	说明
-----	------	----

eid	string	组织 ID, 必填, 取值范围 10000-99999
token	string	登录获取的 token, 必填

### ●Request

POST <http://120.77.254.227/admin/?method=resetEntPassword>

```
{
  "eid": "10000",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ●Response

```
{
  "code": 200, "reason": "success"
}
```

## 5.2.2.6 获取组织属性 (getEntProperty)

### ●parameters

参数名	参数类型	说明
eid	string	组织 ID, 必填
key	string	组织属性 key, 必填
token	string	登录获取的 token, 必填

### key 取值范围:

key (参数)	value (返回值)	说明
"name"	"测试组织 1"	组织名
"contact"	"张三"	联系人
"phone"	"13800001111"	电话
"comment"	"测试"	备注
"pause"	"0"	是否暂停服务(0 为正常, 1 为暂停)

"all_perm"	"0"	已购永久卡数量
"remain_perm"	"0"	剩余永久卡数量
"all_month"	"0"	已购月卡数量
"remain_month"	"0"	剩余月卡数量
"create_time"	"2021-06-27 10:02:59"	创建时间

### ●Request

POST <http://120.77.254.227/admin/?method=getEntProperty>

```
{
  "eid": "10001",
  "key": "name",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDC3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ●Response

```
{
  "code": 200, "reason": "success",
  "value": "测试组织 1",
}
```

## 5.2.2.7 修改组织属性 (setEntProperty)

### ●parameters

参数名	参数类型	说明
eid	string	组织 ID, 必填
key	string	组织属性 key, 必填
value	string	组织属性 value, 必填
token	string	登录获取的 token

**key 取值范围:**

key (参数)	value (参数)	说明
"contact"	李四	联系人
"phone"	13800001111	电话
"comment"	教案	备注
"pause"	1	是否暂停服务(0 为正常, 1 为暂停)
"add_perm"	3	新增永久卡 (填写卡数量)
"add_month"	4	新增月卡 (填写卡数量)

### ●Request

POST <http://120.77.254.227/admin/?method=setEntProperty>

```
{
  "eid": "10001",
  "key": "contact",
  "value": "zcx",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ●Response

```
{
  "code": 200, "reason": "success"
}
```

## 5.3 组织管理员接口说明 (/ent/)

### 5.3.1 组织管理

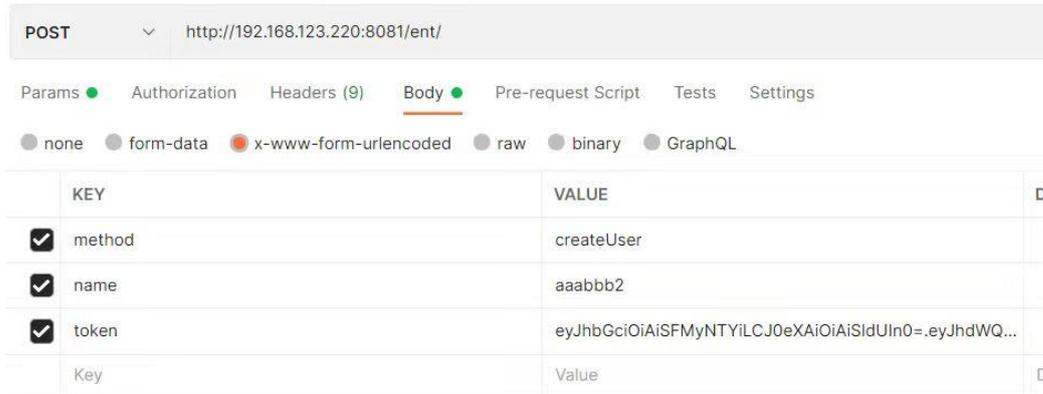
账号 account 为: 10000、10001 等组织 ID (即 eid), 请确保系统管理员 admin 已新建该组织 (即组织 ID 已存在, 否则无法登录)。

默认密码 password 为: 123456

前端调用 login 接口登录成功后获取到 token, 并保留此 token。其它所有接口都必须带上这个 token 作为参数, 用来验证身份和超时 (1 小时)。所有接口调用都采用

POST 方法，并且：

Content-Type 必须设为 application/x-www-form-urlencoded



KEY	VALUE	D
<input checked="" type="checkbox"/> method	createUser	
<input checked="" type="checkbox"/> name	aaabbb2	
<input checked="" type="checkbox"/> token	eyJhbGciOiAiSFMyNTYiLCJ0eXAiOiAiSldUIn0=.eyJhdWQ...	
Key	Value	D

### 5.3.1.1 登录 (login)

#### ●parameters

参数名	参数类型	说明
account	string	账号名（10000 等组织 ID，即 eid），必填
encrypt	string	是否加密（0-不加密，用明文；1-加密），选填，默认 0
password	string	密码，如果加密则先调用 5.4.2 获取加密数据的公钥 getPublicKey，然后用得到的 publicKey 对密码明文进行 RSA 加密。必填，6 位以上明文密码（或其加密结果）。默认不加密、123456。

#### ●Request

POST <http://120.77.254.227/ent/?method=login>

```
{
  "account": "10000",
  "encrypt": "1",
  "password": ""
}
```

#### ●Response

```
{
  "code": 200, "reason": "success",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImlzIjoiIn0="
}
```

```

mV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZX
N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s "
}
    
```

### 5.3.1.2 修改账号密码 (setAccountPassword)

#### ●parameters

参数名	参数类型	说明
encrypt	string	是否加密 (0-不加密, 用明文; 1-加密), 选填, 默认 0
password	string	密码, 如果加密则先调用 5.4.2 获取加密数据的公钥 getPublicKey, 然后用得到的 publicKey 对密码明文进行 RSA 加密。必填, 6 位以上明文密码 (或其加密结果)。默认不加密、123456
token	string	登录获取的 token, 必填

#### ●Request

```

POST http://120.77.254.227/ent/?method=setAccountPassword

{
  "encrypt": "1",
  "password": "",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsI
mV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZX
N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ●Response

```

{
  "code": 200, "reason": "success"
}
    
```

### 5.3.1.3 获取组织属性 (getEntProperty)

#### ●parameters

参数名	参数类型	说明
key	string	组织属性 key, 必填
token	string	登录获取的 token, 必填

**key 取值范围:**

key (参数)	value (返回值)	说明
"name"	"测试组织 1"	组织名
"pause"	"1"	是否暂停服务(0 为正常, 1 为暂停)
"all_perm"	"1"	已购永久卡数量
"remain_perm"	"1"	剩余永久卡数量
"all_month"	"1"	已购月卡数量
"remain_month"	"1"	剩余月卡数量
"create_time"	"2021-06-27 10:02:59"	创建时间
"force_locate"	"1"	是否强制定位 (0 为否, 1 为是)
"record_day"	"1"	服务器数据存储天数 (包括录音、聊天文件、上传位置)
"login_record_day"	"1"	用户上下线记录的保存天数
"listen_channel"	"1"	客户端可监听频道数
"create_channel"	"1"	客户端可创建固定频道数
"manage_channel"	"1"	是否允许客户端管理频道 (0 为否, 1 为是)
"all_user"	"1"	用户总数
"online_user"	"1"	在线用户数
"all_channel "	"1"	固定频道数
"tmp_channel "	"1"	临时频道数

**● Request**

POST <http://120.77.254.227/ent/?method=getEntProperty>

```
{
    "key": "name",
```

```

        "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IjEWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
    }
    
```

● **Response**

```

    {
        "code": 200, "reason": "success",
        "value": "测试组织 1",
    }
    
```

### 5.3.1.4 修改组织属性 (setEntProperty)

● **parameters**

参数名	参数类型	说明
key	string	组织属性 key, 必填
value	string	组织属性 value, 必填
token	string	登录获取的 token, 必填

**key 取值范围:**

key (参数)	value (参数)	说明
" name "	测试组织 1	组织名
" force_locate "	1	是否强制定位 (0 为否, 1 为是)
" record_day "	5	服务器消息存储天数 (包括录音、聊天、位置)
" login_record_day "	20	服务器用户登录记录保存天数
" listen_channel "	3	客户端可监听频道数
" create_channel "	3	客户端可创建固定频道数
" manage_channel "	0	是否允许客户端管理频道 (0 为否, 1 为是)
" all_tmp_call "	0	是否允许非好友间单呼 (0 否, 1 是)

● **Request**



content	string	文字内容，1000 中文、2000 字符内
ttsPlay	string	是否播报语音（0 为否，1 为是）。如果是 0 则终端只显示文字，不播报语音，如果是 1 则终端既显示文字，又播报语音。必填
token	string	登录获取的 token，必填

### ●Request

POST <http://120.77.254.227/ent/?method=pushNotice>

```
{
  "targetType": "1 ",
  "targetId": "1001 ",
  "content": " 频道全体集合 ",
  "ttsPlay": 1,
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

### ●Response

```
{
  "code": 200,"reason": "success"
}
```

## 5.3.2 用户管理（创建、删除、获取、修改）

### 5.3.2.1 新建用户（createUser）

#### ●parameters

参数名	参数类型	说明
name	string	用户名（只能是字符或数字，无空格、不能是 7 位数字），必填
cardType	string	卡类型（perm-永久卡，month-月卡）。选填，默认"perm"



**POST** <http://120.77.254.227/ent/?method=createUsersInBatch>

```
{
  "users": [{"name":" nameA500","nick":"nickA500","cardType":"perm","cardCount":"2"},
  {"name":" nameA501","nick":"nickA501","cardType":"perm","cardCount":"2"},
  {"name":" nameA502","nick":"nickA502","cardType":"perm","cardCount":"2"}],
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
```

● **Response**

```
{
  "code": 200, "reason": "success",
  "successCount":100 //成功新建用户的数量
}
```

### 5.3.2.3 删除用户 (deleteUser)

● **parameters**

参数名	参数类型	说明
uid	string	用户 ID, 必填
token	string	登录获取的 token, 必填

● **Request**

**POST** <http://120.77.254.227/ent/?method=deleteUser>

```
{
  "uid":"1001001",
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
```

● **Response**

```
{
    "code": 200, "reason": "success"
}
```

### 5.3.2.4 获取用户总数 (getUsersTotal)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填

#### ●Request

POST <http://120.77.254.227/ent/?method=getUsersTotal>

```
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNzc3OTkwMzkyLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXM0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
    "code": 200, "reason": "success",
    "data": 9
}
```

### 5.3.2.5 分页获取用户列表 (getUsersInPage)

#### ●parameters

参数名	参数类型	说明
pageIndex	int	页面索引, 必填, 从 0 开始
pageSize	int	页面用户数, 必填
filter	string	过滤条件 (过滤格式: "role=1;ban=1;card=month;online=0")。选填, 每个过滤条件都选填

keyword	string	搜索关键词（包含搜索用户 ID、用户名、用户昵称），选填
token	string	登录获取的 token，必填

●Request

POST <http://120.77.254.227/ent/?method=getUsersInPage>

```
{
  "pageIndex":0,
  "pageSize":20,
  "filter":" ban=0",
  "keyword": "",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

●Response

```
{
  "code":200,"reason":"success",
  "total": 2, //搜索结果总数，而不是分页结果数量
  "data":[{"ban": 0,"cardType": "month","cid": -1,"device": "", "lastLogin": "", "name": "11100000675","nick": "TT75","regTime": "2021-06-27 10:05:15","role": 0,"timeout": "", "uid": 1001035}, {"ban": 0,"cardType": "month","cid": -1,"device": "", "lastLogin": "", "name": "11100000676","nick": "TT76","regTime": "2021-06-27 10:05:15","role": 0,"timeout": "", "uid": 1001036,"joinedChannels": "频道 1(1001);频道 2(1002);"}] //ban: 是否黑名单 (0 为否, 1 为是)

  //role: 角色 (0 为普通, 1 为调度员)
  // cardType: 卡类型 (perm 为永久卡, moth 为月卡)
  // joinedChannels: 已加入的频道 (多个频道分号隔开, 每个频道格式为 <频道名>(频道 ID) )
}
```

### 5.3.2.6 重置用户密码 (resetUserPassword)

#### ●parameters

参数名	参数类型	说明
uid	string	用户 ID, 必填
token	string	登录获取的 token, 必填

#### ●Request

```
POST http://120.77.254.227/admin/?method=resetUserPassword
{
  "uid": "1001016",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200, "reason": "success"
}
```

### 5.3.2.7 获取用户属性 (getUserProperty)

#### ●parameters

参数名	参数类型	说明
uid	string	用户 ID, 必填
key	string	用户属性 key, 必填
token	string	登录获取的 token, 必填

#### key 取值范围:

key (参数)	value (返回值)	说明
"name"	"测试用户 1"	用户名

"nick"	"测试昵称 1"	用户昵称
"uid"	"1001036"	用户 ID
"cid"	"1001"	当前所在频道 id。<0 表示离线,0 表示在守候频道(2025.03.05 后)
"device"	"huawei P30"	上次登录设备
"lastLogin"	"2021-06-27 10:05:15"	上次登录时间
"regTime"	"2021-06-27 10:05:15"	注册时间
"timeout"	"2021-06-27 10:05:15"	过期时间
"joined_channels"	"xyz(1001);ab(1018);"	已加入频道
"cardType"	"perm"	用户卡类型 (perm 永久卡, month 月卡)
"ban"	"1"	用户是否黑名单 (0 为否, 1 为是)
"role"	"1"	用户角色 (0 为普通, 1 为调度员)

#### ●Request

POST <http://120.77.254.227/ent/?method=getUserProperty>

```
{
  "uid": "1001035",
  "key": "name",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200, "reason": "success",
  "value": "测试用户 1",
}
```

### 5.3.2.8 修改用户属性 (setUserProperty)

#### ●parameters

参数名	参数类型	说明
uid	string	用户 ID, 必填
key	string	用户属性 key, 必填
value	string	用户属性 value, 必填
token	string	登录获取的 token, 必填

**key 取值范围:**

key (参数)	value (参数)	说明
" name "	测试用户 1	用户名
" nick "	测试 1	用户昵称
" cardType "	perm	用户卡类型 (perm 永久卡, month 月卡)
" addMonth "	5	增加月卡数量
" ban "	1	用户是否黑名单 (0 为否, 1 为是)
" role "	1	用户角色 (0 为普通, 1 为调度员)

**●Request**

POST <http://120.77.254.227/ent/?method=setUserProperty>

```
{
  "uid": "1001001",
  "key": " password ",
  "value": "test ", //6 位以上
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE5OTkxMzQ3OTkwMzZlcjE1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200, "reason": "success"
}
```

### 5.3.2.9 所有用户批量增加永久卡 (batchAddPerm)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填

#### ●Request

```
POST http://120.77.254.227/ent/?method=batchAddPerm
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzgzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
    "code": 200, "reason": "success"
}
```

注：为所有用户补齐永久卡，已有永久卡者将忽略，用户剩余月卡回收到组织。如果组织剩余永久卡不足，则在永久卡用完时终止操作，已赋卡用户继续生效。

### 5.3.2.10 所有用户批量增加月卡 (batchAddMonth)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填
cardCount	string	要增加的月卡数量, 选填, 默认 1

#### ●Request

```
POST http://120.77.254.227/ent/?method=batchAddMonth
{
    "cardCount": "3",
}
```

```

    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
  }
  
```

● **Response**

```

  {
    "code": 200, "reason": "success"
  }
  
```

注：为所有用户增加指定数量的月卡，已有永久卡者将忽略。如果组织剩余月卡不足，则在月卡用完时终止操作，已赋卡用户继续生效。

### 5.3.2.11 所有用户补足若干月内服务 (batchAddToMonth)

● **parameters**

参数名	参数类型	说明
token	string	登录获取的 token，必填
toMonth	string	要补足的月数量。选填，默认 1。例如要保证所有用户一个月内服务时长，则 toMonth 取值"1"

● **Request**

```

POST http://120.77.254.227/ent/?method=batchAddToMonth

{
  "toMonth": "3",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
  
```

● **Response**

```

  {
    "code": 200, "reason": "success"
  }
  
```

```
}

```

注：为所有用户补足若干个月内服务时长，已有足够服务时长者将忽略。如果组织剩余月卡不足，则在月卡用完时终止操作，已赋卡用户继续生效。

### 5.3.2.12 分页获取用户上下线记录 (getLoginRecordInPage)

#### ●parameters

参数名	参数类型	说明
uid	string	用户 ID，必填
pageIndex	int	页码，从 0 开始。必填
pageSize	int	每页记录的数量。必填
token	string	登录获取的 token，必填

#### ●Request

**POST** http://120.77.254.227/ent/?method=getLoginRecordInPage

```
{
  "uid": "1001003",
  "pageIndex": 0,
  "pageSize": 50,
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE5MzQ3OTkwMzZLCj1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200, "reason": "success",
  "total": 120, //搜索结果总数，而不是分页结果数量
  "data": [{"time": "2023-04-12 15:16:13", "online": true, "device": "华为 Mate10"},
           {"time": "2023-04-12 18:12:33", "online": false, "device": ""}]
}
```



```
mV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZX
N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
```

### ● Response

```
{
  "code": 200,"reason": "success",
  "value":"20",
}
```

注：目前服务器支持 3 个月内用户使用统计。

## 5.3.3 频道管理（创建、删除、获取、修改）

### 5.3.3.1 新建频道（createChannel）

#### ● parameters

参数名	参数类型	说明
name	string	频道名，必填
cid	string	频道 ID，选填，一般不需要
token	string	登录获取的 token，必填

#### ● Request

```
POST http://120.77.254.227/ent/?method=createChannel
{
  "name":"test",
  "cid":"1006",
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsI
mV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZX
N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
```

#### ● Response

```
{
```



```

        "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
    }
    
```

● **Response**

```

    {
        "code": 200, "reason": "success",
        "data": 9
    }
    
```

### 5.3.3.4 分页获取频道列表 (getChannelsInPage)

● **parameters**

参数名	参数类型	说明
pageIndex	int	页码，必填，从 0 开始
pageSize	int	每页数量，必填
keyword	string	搜索关键词（包含搜索频道 ID、频道名），选填
token	string	登录获取的 token，必填

● **Request**

```

POST http://120.77.254.227/ent/?method=getChannelsInPage
{
    "pageIndex": 1,
    "pageSize": 20,
    "keyword": "",
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

● **Response**

```
{
  "code":200,"reason":"success",
  "total": 2, //搜索结果总数，而不是分页结果数量
  "data":[{"cid":1001,"memberNum":13,"name": "频道 1"},
         {"cid": 1002,"memberNum": 5,"name": "频道 2"}]
}
```

### 5.3.3.5 修改频道名称 (setChannelName)

#### ●parameters

参数名	参数类型	说明
cid	string	频道 ID，必填
name	string	新频道名，必填
token	string	登录获取的 token，必填

#### ●Request

POST <http://120.77.254.227/ent/?method=setChannelName>

```
{
  "cid":"1001",
  "name":"test",
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200,"reason": "success"
}
```

### 5.3.3.6 获取频道属性 (getChannelProperty)

#### ●parameters



**●parameters**

参数名	参数类型	说明
cid	string	频道 ID, 必填
key	string	键, 必填
value	string	值, 必填
token	string	登录获取的 token, 必填

**●Request**

POST <http://120.77.254.227/ent/?method=setChannelProperty>

```
{
  "cid": "1001",
  "key": "reserved",
  "value": "1",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**key 取值范围:**

key (参数)	value (参数)	说明
"name"	"测试频道 2"	频道名
"reserved"	"1"	保留频道。设置 0 不保留, 1 保留。 设置保留后, 禁止客户端删除频道

**●Response**

```
{
  "code": 200, "reason": "success"
}
```

### 5.3.4 频道成员管理 (添加、移除、获取、修改)

#### 5.3.4.1 添加频道成员 (addMember)

**●parameters**

参数名	参数类型	说明
cid	string	频道 ID, 必填
membersId	string	成员 id 表, 以逗号分隔, 必填
token	string	登录获取的 token, 必填

**●Request**

POST <http://120.77.254.227/ent/?method=addMember>

```
{
  "cid": "1001",
  "membersId": "1001001,1001002",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200, "reason": "success"
}
```

### 5.3.4.2 移除频道成员 (removeMember)

**●parameters**

参数名	参数类型	说明
cid	string	频道 ID, 必填
membersId	string	成员 id 表, 以逗号分隔, 必填
token	string	登录获取的 token, 必填

**●Request**

POST <http://120.77.254.227/ent/?method=removeMember>

```
{
  "cid": "1001",
```

```

    "membersId":"1001001,1001002",
    "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ●Response

```

{
    "code": 200,"reason": "success"
}
    
```

### 5.3.4.3 获取频道成员总数 (getMembersTotal)

#### ●parameters

参数名	参数类型	说明
cid	string	频道 ID, 必填
token	string	登录获取的 token, 必填

#### ●Request

**POST** <http://120.77.254.227/ent/?method=getMembersTotal>

```

{
    "cid":"1004",
    "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ●Response

```

{
    "code":200,"reason": " success",
    "data":9
}
    
```



参数名	参数类型	说明
token	string	登录获取的 token, 必填
cid	string	频道 id, 必填
uid	string	用户 id, 必填

●Request

```
POST http://192.168.123.49/ent/?method=getMemberRole

{
  "cid": "1001",
  "uid": "1001001",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzgzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

●Response

```
{
  "code": 200, "reason": "success",
  "data": 1 // (0-普通, 1-副管理员, 2-管理员)
}
```

### 5.3.4.6 修改频道成员身份 (setMemberRole)

●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填
cid	string	频道 id, 必填
uid	string	用户 id, 必填
role	string	成员身份, 必填, (0-普通, 1-副管理员, 2-管理员)

●Request

```
POST http://192.168.123.49/ent/?method=setMemberRole

{
```

```

"cid":"1001",

"uid":"1001001",

"role":"1",

"token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IndleWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"

}
    
```

●Response

```

{

  "code": 200,"reason": "success",

}
    
```

### 5.3.4.7 获取频道成员麦权级别 (getMemberLevel)

●parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填
cid	string	频道 id，必填
uid	string	用户 id，必填

●Request

```

POST http://192.168.123.49/ent/?method=getMemberLevel

{

  "cid":"1001",

  "uid":"1001001",

  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IndleWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"

}
    
```

●Response

```

{
    
```



**● parameters**

参数名	参数类型	说明
pageIndex	Int	页码, <b>必填</b> (从 0 开始)
pageSize	Int	每页数量, <b>必填</b>
uid	string	用户 ID, 选填 (不填则为所有用户)
cid	string	频道 ID, 选填 (不填则为所有频道)
startDate	string	开始时间 (yyyy-MM-dd 或 yyyy/MM/dd), 选填 (不填则不限制开始时间)
endDate	string	结束时间 (yyyy-MM-dd 或 yyyy/MM/dd), 选填 (不填的话不限制结束时间)
token	string	登录获取的 token, <b>必填</b>

**● Request**

POST <http://120.77.254.227/ent/?method=getVoicesInPage>

```
{
  "pageIndex":0,
  "pageSize":20,
  "uid":" 1001015 ",
  "cid":" 1001 ",
  "startDate":"2021-10-01",
  "endDate":"2021-10-06",
  "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**● Response**

```
{
  "code": 200,"reason": "success","total": 2, //搜索结果总数, 而不是分页结果数量
  "data":[{"cid": 1001,"duration": 1,"uid": 1001015,"time":"2021-10-04 07:30:09","url":
```



```

        "code": 200,"reason": "success",

        "url":"http://192.168.123.69:59680/TotalkWar/downloadVoiceZip.do?token=eyJhbGciOiAiS
        FMyNTYiLCJ0eXAiOiAiSldUIn0=.eyJhdWQiOiAiMTAwMDAiLCJleHAiOiAiMTYzNDAzMD
        gyOCJ9.MeHARu0uUNaBKQxjvVwCsnKpgKsrtVB7r60Tddeyl0=&eid=10000&cid=-1&uid=-1
        &startDate=&endDate="           //打包下载录音地址
    }
    
```

### 5.3.5.3 分页获取频道消息列表 (getMessagesInPage)

#### ● parameters

参数名	参数类型	说明
pageIndex	int	页码, <b>必填</b> (从 0 开始)
pageSize	int	每页数量, <b>必填</b>
cid	string	频道 ID, <b>必填</b>
sort	string	时间排序方式, 选填, 取值 down 或 up, 默认 down(降序)
token	string	登录获取的 token, <b>必填</b>

#### ● Request

POST <http://120.77.254.227/ent/?method=getMessagesInPage>

```

{
    "pageIndex":0,
    "pageSize":20,
    "cid": " 1001 ",
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsI
    mV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkyLCJ1c2VybmFtZSI6ImV4cCI6MTQ3ODU5NTE5N0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ● Response

```

{
    "code": 200,"reason": "success","total": 2,           //搜索结果总数, 而不是分页结果数量
}
    
```

```

        "data":[{"cid":      1001,"chan_seq_id":      123,"from_id":      1001015,"time":
16732800000339,"type":1,content:  "  你  好  ",orig_name:  "",size:0,duration:0},{ "cid":
1001,"chan_seq_id": 124,"from_id": 1001015,"time": 16732800000300,"type":1,content: "你好
",orig_name: "",size:0,duration:0},]
    }
    
```

说明:

返回结果默认时间降序排列

type: 0-文字 1-语音 2-图片 3-短视频 4-直播 5-位置 6-文件, 7-抢单

content: 对于文字消息, 就是消息文本; 对于语音、图片、短视频等媒体, 是媒体文件在服务器上存储的相对路径; 对于位置消息, 是位置消息详细信息

orig\_name: 仅对 6-文件类型消息有效, 表示文件的原始名称

size: 仅对图片、短视频、文件消息有效, 表示媒体文件的字节数

duration: 仅对语音、短视频、直播消息有效, 表示时长秒数

## 5.3.6 运营统计

### 5.3.6.1 频道人数排行 (getChannelRanks)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填

#### ●Request

POST <http://120.77.254.227/ent/?method=getChannelRanks>

```

{
    "token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNzc3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ●Response

```
{
```

```

"code": 200,"reason": "success",

//频道人数排名（最多 10 个频道）

"data": [{"chan_id": 1001,"chan_name": "test2","user_count": 89},
{"chan_id": 1003,"chan_name": "test4","user_count": 73},
.....
{"chan_id": 1002,"chan_name": "test5","user_count": 45}]
    
```

### 5.3.6.2 24 小时内每小时在线人数 (getOnlineNumbers)

#### ● parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填

#### ● Request

```

POST http://120.77.254.227/ent/?method=getOnlineNumbers

{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ● Response

```

{
    "code": 200,"reason": "success",

// number 为在线人数（一共 24 个时间点）

"data": [{"number": 5,"time": "17:31:34"},
{"number": 5,"time": "18:31:34"},
.....
{"number": 10,"time": "23:31:34"}]
}
    
```

### 5.3.6.3 最近若干天每日新增人数 (getIncreaseNumbers)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token, 必填
startDate	string	查询开始日, 格式形如 2021-12-03 或 2021/12/03。选填。 如果不填, 默认为 7 天前
endDate	string	查询结束日, 格式形如 2021-12-03 或 2021/12/03。选填。 如果不填, 默认为现在

#### ●Request

```
POST http://120.77.254.227/ent/?method=getIncreaseNumbers
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE5MzQ3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
    "code": 200, "reason": "success",
    "data": [{"day": "2021-12-20", "number": 0}, // number 为当日新增人数
            {"day": "2021-12-21", "number": 0},
            {"day": "2021-12-22", "number": 0},
            {"day": "2021-12-23", "number": 0},
            {"day": "2021-12-24", "number": 0},
            {"day": "2021-12-25", "number": 0},
            {"day": "2021-12-26", "number": 0}]
}
```

### 5.3.6.4 24 小时内登录人数 (getActiveNumber)

**●parameters**

参数名	参数类型	说明
token	string	登录获取的 token, 必填

**●Request**

POST <http://120.77.254.227/ent/?method=getActiveNumber>

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzgzLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200, "reason": "success",
  "data": 0 //24 小时内登录人数
}
```

### 5.3.6.5 当前正在呼叫人数 (getCallingNumber)

**●parameters**

参数名	参数类型	说明
token	string	登录获取的 token, 必填

**●Request**

POST <http://120.77.254.227/ent/?method=getCallingNumber>

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzgzLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200,"reason": "success",
  "data": 0 //当前正在呼叫人数
}
```

### 5.3.6.6 当前正在频道内接收语音路数 (getReceivingNumber)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填

#### ●Request

POST <http://120.77.254.227/ent/?method=getReceivingNumber>

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200,"reason": "success",
  "data": 0 //当前正在频道内接收语音路数
}
```

### 5.3.6.7 监听语音路数 (getListeningNumber)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填

#### ●Request

```
POST http://120.77.254.227/ent/?method=getListeningNumber
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

● Response

```
{
    "code": 200, "reason": "success",
    "data": 0 //当前监听语音路数（不在频道里，但监听语音。此数据可能比实际值大，仅供参考）
}
```

### 5.3.6.8 最近若干天每日呼叫总次数 (getCallNumbers)

● parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填
startDate	string	查询开始日，格式形如 2021-12-03 或 2021/12/03。选填。如果不填，默认为 7 天前
endDate	string	查询结束日，格式形如 2021-12-03 或 2021/12/03。选填。如果不填，默认为现在

● Request

```
POST http://120.77.254.227/ent/?method=getCallNumbers
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkzLCJ1c2VybmFtZSI6IjZSI6Ind1eWF0ZXN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200,"reason": "success",
  "data": [{"day": "2021-12-20","number": 29}, // number 为当日呼叫总次数
    {"day": "2021-12-21","number": 0},
    {"day": "2021-12-22","number": 7},
    {"day": "2021-12-23","number": 0},
    {"day": "2021-12-24","number": 0},
    {"day": "2021-12-25","number": 0},
    {"day": "2021-12-26","number": 0}]
}
```

### 5.3.6.9 最近若干天每日呼叫总时长 (getCallDurations)

**●parameters**

参数名	参数类型	说明
token	string	登录获取的 token，必填
startDate	string	查询开始日，格式形如 2021-12-03 或 2021/12/03。选填。 如果不填，默认为 7 天前
endDate	string	查询结束日，格式形如 2021-12-03 或 2021/12/03。选填。 如果不填，默认为现在

**●Request**

**POST** <http://120.77.254.227/ent/?method=getCallDurations>

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDE3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

**●Response**

```
{
  "code": 200,"reason": "success",
  "data": [{"day": "2021-12-20","number": 30}, //number 为当日呼叫总时长（秒数）
  {"day": "2021-12-21","number": 0},
  {"day": "2021-12-22","number": 11},
  {"day": "2021-12-23","number": 0},
  {"day": "2021-12-24","number": 0},
  {"day": "2021-12-25","number": 0},
  {"day": "2021-12-26","number": 0}]
}
```

### 5.3.6.10 最近若干天每日活跃用户 (getMultiActive)

#### ●parameters

参数名	参数类型	说明
token	string	登录获取的 token，必填
startDate	string	查询开始日，格式形如 2021-12-03 或 2021/12/03。 <b>选填。</b> 如果不填，默认为 7 天前
endDate	string	查询结束日，格式形如 2021-12-03 或 2021/12/03。 <b>选填。</b> 如果不填，默认为现在

#### ●Request

POST http://120.77.254.227/ent/?method=getMultiActive

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
```

#### ●Response

```
{
  "code": 200,"reason": "success",
  "data": [{"day": "2021-12-20","number": 30}, //number 为当日活跃用户数
}
```

```

        {"day": "2021-12-21","number": 0},
        {"day": "2021-12-22","number": 11},
        {"day": "2021-12-23","number": 0},
        {"day": "2021-12-24","number": 0},
        {"day": "2021-12-25","number": 0},
        {"day": "2021-12-26","number": 0}]
    }
    
```

## 5.4 通用接口说明 (/system/)

### 5.4.1 获取 token 的账号名 (getAccountByToken)

#### ●parameters

参数名	参数类型	说明
token	string	token, 必填

#### ●Request

POST <http://192.168.123.49/system/?method=getAccountByToken>

```

{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6IjEyMzQ1NiIsImV4cCI6MTQ3ODU5NTE5MywiaWF0IjoxNDc3OTkwMzkyLCJ1c2VybmFtZSI6ImN0MSJ9.UA-6u6ryC-VfuoMkg3nvtrNWYS6hohfhZzFjDsy072s"
}
    
```

#### ●Response

```

{
    "code": 200, "reason": "success",
    "account": 10000
}
    
```

### 5.4.2 获取加密数据的公钥 (getPublicKey)

#### ●parameters

参数名	参数类型	说明
无	无	无

●Request

POST <http://192.168.123.49/system/?method=getPublicKey>

```
{
}
```

●Response

```
{
  "code": 200,"reason": "success",
  "publicKey":"-----BEGINPUBLICKEY-----\nMIGfMA0GCSqGSIb3DQEBAQUAA4GNAD
  CBiQKBgQCh/8/kU4Um2FUrkTYXWQDW4MqY\nuiQQ1WaEtL4JQSy6w64iqWoBDPcS+BF
  haPj8gr1Da+F9Up/s7CSj4Do5h3kMHmRf\nXu2Ba/Hxczf/QB+Cutm386roP41bAKolkUVgWIZ
  EDjOc4+sqrPygxDa0dtbugrdS\nvJXqUcmJSIkVnVXj0QIDAQAB\n-----ENDPUBLICKEY-----"
}
```

## 6 用户位置及轨迹查询接口

见文档《位置及轨迹查询接口定义》。