# 跨界云通IM即时通信开发手册 V1.3

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 作者 | 版本 | 备注 |
| 2020-9-28 | openbpi,apowners | V1.3 | 跨界云通v1.3提供跨界单聊和跨界群聊功能。 |
|  |  |  |  |
|  |  |  |  |

## 简介

跨界云通 是一套IM即时通信的解决方案，具备基础的单聊、群聊、聊天室的功能，此外还实现了开放式社交公共网络中的OSCP协议，使用跨界云通，企业可以直接接入OSPN网络，与其他APP之间进行互通。

## 功能介绍

### 跨界单聊

跨界云通除了提供普通的单聊功能以外，还提供支持开放式社交通信协议（OSCP）的跨界单聊功能，可以让企业的APP和其他APP的用户进行聊天。

#### 架构介绍



1. 用户的通信账号由应用服务器注册以后发送给移动端用户
2. 移动端用户登录IMS
3. 移动端用户发送消息给IMS，IMS通过connector转发到其他企业的服务器上。

### 跨界群

跨界云通除了提供普通的群聊功能以外，还提供跨界群的功能，可以让不同APP的用户在同一个群里聊天。

#### 架构介绍



该框架图展示了不同企业的用户如何在同一个群里聊天。

用户A和用户D在同一企业，可以直接通过群服务器CGS进行通信。

用户B与用户C不在同一企业，他们会先将消息发送给自己所在企业的单聊服务器。

由单聊服务器转发到CGS所在的单聊服务器，然后再转发给CGS。

CGS收到消息以后，通过单聊服务器，转发给所有人。

## 安装与部署

java部署

1. java部署环境要求：java，MySQL 8.0
2. 修改配置文件

输入命令 vi ospn.properties

配置注册密码：adminKey=

配置IMS的IP：ipIMServer=当前公共IP

配置connector的IP：ipConnector=当前公共IP

配置已知在线节点：ipPeer=任一在线节点IP

保存并退出

配置mysql数据库

输入命令vi c3p0-config.xml

<property name="jdbcUrl">jdbc:mysql://localhost:3306/dbname?useSSL=false&amp;serverTimezone=GMT%2B8&amp;allowPublicKeyRetrieval=true&amp;useUnicode=true&amp;characterEncoding=UTF-8</property>

<property name="user">user</property>

<property name="password">password</property>

配置以上红色部分

localhost:3306为数据库ip地址和端口

dbname为数据库的名字

user为连接数据库的账户

password为修改连接数据库账户的密码

保存并退出。

1. 使用start.sh启动服务器

docker部署

1. 环境要求：docker，MySQL 8.0
2. 下载docker镜像

输入命令 docker pull registry.cn-zhangjiakou.aliyuncs.com/ospn-hub/ims:v1.3

1. 启动镜像

输入命令

docker run -itd --name=ims -p 8100:8100 -p 8200:8200 -p 8300:8300 -p 8400:8400 -p 8500:8500 -p 8600:8600 -p 8700:8700 registry.cn-zhangjiakou.aliyuncs.com/ospn-hub/ims:v1.3 bash bash

1. 修改配置文件

输入命令 docker attack ims

修改配置文件请参考java部署的修改配置文件。

1. 启动服务

输入命令./start.sh

### 防火墙

防火墙必须开放8500以及映射的以下端口

8100

8200

8300

8400

8600

## 接入OSPN

企业可以决定是否接入OSPN网络，接入了OSPN网络以后，用户可以和其他接入的用户进行跨界聊天。不接入则只能在自己的服务器内聊天。

接入OSPN的配置

ipPeer=10.10.10.10

将ipPeer设置成OSPN中的任意一个可用节点以后，重启服务即可。

接入以后，企业仍然有选择断开OSPN的权利。

断开OSPN网络：将ipPeer设置为一个不可用节点以后重启服务。

## 服务器端接口



应用服务器通过POST将用户注册信息发送到IM服务器，完成用户注册操作

url：<http://imServer:port/admin>

imServer为IMS的IP。

port固定为8300

下文中使用的密码（password）为配置文件中的adminKey。

注册信息为json格式：

{“username”:”testname”, “password”:”123456”}

以上信息需使用 AES/CBC/PKCS5Padding 算法加密

* Aes的密钥为明文密码通过sha256算法得到，此明文密码要跟IM配置文件里面的adminKey配置项匹配
* Aes加密后的数据以Base58格式编码

最终发送到IM服务器的数据格式为：

{“command”:”register”, “data”:”Base58 encode data”}

IM服务器返回信息{“errCode”:”success”}表示注册成功，其他表示失败

## 客户端接口

### Android SDK

#### 使用接口

##### 初始化sdk

public void initSDK (Context context, String ip, OSNListener listener)

说明: 初始化sdk

参数:

|  |  |
| --- | --- |
| context | 上下文 |
| ip | 服务器ip地址,格式:”xxx.xxx.xxx.xxx:xxxx” |
| listener | SDK的回调 |

返回值: void

##### 重置服务器IP

public void resetHost(String ip)

说明: 重置服务器IP

参数:

|  |  |
| --- | --- |
| ip | 服务器ip地址,格式:”xxx.xxx.xxx.xxx:xxxx” |

返回: void

##### 登录

public boolean login (String userID, OSNGeneralCallback callback)

说明: 使用用户ID登录

参数:

|  |  |
| --- | --- |
| userID | 用户ID |
| callback | 回调 |

返回: 登录成功返回true，失败返回false

public boolean login(String userName, String password, OSNGeneralCallback callback)

说明: 使用用户用户名密码登录

参数:

|  |  |
| --- | --- |
| userName | 用户名 |
| Password | 用户密码 |
| callback | 回调 |

返回: 登录成功返回true，失败返回false

##### 登出

public void logout (OSNGeneralCallback callback)

说明: 登出

参数:

|  |  |
| --- | --- |
| callback | 回调 |

返回: void

##### 获取当前用户ID

public String getUserID()

说明: 获取当前用户ID

参数:

返回: 当前用户ID

##### 获取用户信息

public UserInfo getUserInfo(String userID)

说明：获取用户信息

参数：

|  |  |
| --- | --- |
| userID | 用户ID |

返回: 返回用户信息

##### 获取群组信息

public GroupInfo getGroupInfo(String userID)

说明：获取群组信息

参数：

|  |  |
| --- | --- |
| groupID | 群组ID |

返回: 返回群组信息

##### 修改用户信息

public void modifyUserInfo(List<String> keys, UserInfo userInfo, OSNGeneralCallback callback)

说明：修改用户信息

参数：

|  |  |
| --- | --- |
| keys | 需要修改的信息列表，比如”name”，”portrait”等 |
| userInfo | 待修改的用户信息，keys列表当中的项对应的数据必须有效 |
| callback | 回调 |

返回: void

##### 获取当前朋友信息列表

public List<FriendInfo> getFriendList()

说明: 获取当前朋友信息列表

参数

返回: 当前朋友信息列表(注:需要说明返回的数据结构)

##### 邀请朋友

public void inviteFriend (String userID, String reason, OSNGeneralCallback callback)

说明: 邀请朋友

参数

|  |  |
| --- | --- |
| userID | 朋友用户ID |
| reason | 打招呼信息 |
| callback | 回调 |

返回: void

##### 删除好友

public void deleteFriend (String userID, OSNGeneralCallback callback)

说明: 删除朋友

参数:

|  |  |
| --- | --- |
| userID | 朋友用户ID |
| callback | 回调 |

返回: void

##### 同意邀请

public void acceptFriend (String userID, OSNGeneralCallback callback)

说明: 同意邀请

参数:

|  |  |
| --- | --- |
| userID | 朋友用户ID |
| callback | 回调 |

返回: void

##### 发送文本消息

public void sendTextMessage (String text, String userID, OSNGeneralCallback callback)

说明: 发送文本消息

参数:

|  |  |
| --- | --- |
| text | 消息内容 |
| userID | 目标用户ID |
| callback | 回调 |

返回: void

##### 上传数据

public String uploadData(String fileName, byte[] data, OSNTransferCallback callback)

说明: 上传数据

参数

|  |  |
| --- | --- |
| fileName | 文件名 |
| data | 数据 |
| callback | 回调，数据格式为json，包含数据在服务器的url，例{“url”,”urlData”} |

返回: 数据在服务器的url

##### 下载数据

public void downloadData(String remoteUrl, String localPath, String target, OSNTransferCallback callback)

说明: 下载数据

参数

|  |  |
| --- | --- |
| remoteUrl | 数据在服务器的url，由上传数据接口返回 |
| localPath | 数据下载的本地路径 |
| target | 拥有此数据的osnID，比如用户的头像，此数据为该用户的osnID |
| callback | 回调 |

返回: void

#### 跨界群SDK

##### 创建群组

public void createGroup(String groupName List<String> member, String portrait, OSNGeneralCallback callback)

说明: 创建群组

参数

|  |  |
| --- | --- |
| groupName | 群组名字 |
| member | 群组初始成员 |
| portrait | 群组头像url，由上传数据接口返回 |
| callback | 回调 |

返回: void

##### 添加成员

public void addMember (String groupID, List<String> members, OSNGeneralCallback callback)

说明: 添加群组成员

参数

|  |  |
| --- | --- |
| groupID | 群组ID |
| Members | 成员osnID列表 |
| callback | 回调 |

返回: void

##### 删除成员

public void delMember (String groupID, List<String> members, OSNGeneralCallback callback)

说明: 添加群组成员

参数

|  |  |
| --- | --- |
| groupID | 群组ID |
| Members | 成员osnID列表 |
| callback | 回调 |

返回: void

##### 退出群组

public void quitGroup (String groupID, OSNGeneralCallback callback)

说明: 退出群组

参数

|  |  |
| --- | --- |
| groupID | 群组ID |
| callback | 回调 |

返回: void

##### 解散群组

public void dismissGroup (String groupID, OSNGeneralCallback callback)

说明: 解散群组

参数

|  |  |
| --- | --- |
| groupID | 群组ID |
| callback | 回调 |

返回: void

##### 修改群组信息

public void modifyGroupInfo(List<String> keys, GroupInfo groupInfo, OSNGeneralCallback callback)

说明: 修改群组信息

参数

|  |  |
| --- | --- |
| Keys | 需要修改的信息列表，比如”name”，”portrait”等 |
| groupInfo | 待修改的群组信息，keys列表当中的项对应的数据必须有效 |
| callback | 回调 |

返回: void

#### 数据结构

##### osnsdk事件回调接口

**public interface** OSNListener {  
 **void** onConnectSuccess ();  
 **void** onConnectFailed (String error);  
 **void** onRecvTextMessage (List<MessageInfo> msgList);  
 **void** onFriendRequest(FriendRequest request);  
 **void** onFriendUpdate(List<FriendInfo> userIDList);  
 **void** onUserUpdate(UserInfo userInfo, List<String> keys);  
 **void** onGroupUpdate(String state, GroupInfo groupInfo, List<String> keys);  
}

###### 连接服务器成功

**void** onConnectSuccess ();

说明: 连接服务器成功

参数

返回: void

###### 连接服务器失败

**void** onConnectFailed (String error);

说明: 连接服务器失败

参数

|  |  |
| --- | --- |
| Error | 失败信息 |

返回: void

###### 接收文本消息

**void** onRecvTextMessage (List<MessageInfo> msgList);

说明: 接收文本消息

参数

|  |  |
| --- | --- |
| msgList | 消息列表 |

返回: void

###### 添加朋友请求

**void** onFriendRequest(FriendRequest request);

说明: 添加朋友请求

参数

|  |  |
| --- | --- |
| Request | 请求数据 |

返回: void

###### 朋友信息更新

**void** onFriendUpdate(List<FriendInfo> userIDList);

说明: 朋友信息更新，比如添加、删除朋友

参数

|  |  |
| --- | --- |
| Request | 请求数据 |

返回: void

###### 用户信息更新

**void** onUserUpdate(UserInfo userInfo, List<String> keys);

说明: 用户信息更新

参数

|  |  |
| --- | --- |
| userInfo | 更新的用户信息，keys包含的项必须有效 |
| Keys | 需要更新的信息列表，比如”name”，”portrait”等 |

返回: void

###### 群组信息更新

**void** onGroupUpdate(String state, GroupInfo groupInfo, List<String> keys);

说明: 群组信息更新

参数

|  |  |
| --- | --- |
| State | 更新操作状态，比如”AddMember”，”CreateGroup”等 |
| groupInfo | 更新的群组信息，keys包含的项必须有效 |
| Keys | 需要更新的信息列表，比如”name”，”portrait”等 |

返回: void

##### Osnsdk操作回调

**public interface** OSNGeneralCallback {  
 **void** onSuccess(String json);  
 **void** onFailure(String error);  
}

###### 操作成功

**void** onSuccess(String json);

说明: 操作成功

参数

|  |  |
| --- | --- |
| Json | Json格式返回数据，具体数据参考各个接口 |

返回: void

###### 操作失败

**void** onFailure(String error);

说明: 操作失败

参数

|  |  |
| --- | --- |
| error | 失败信息 |

返回: void

##### Osnsdk数据类

**public class** FriendInfo {  
 **public** String **userID**;  
 **public** String **friendID**;  
 **public int state**; *//"0, normal; 1, deleted; 2, blacked"* **public static final int *Normal*** = 0;  
 **public static final int *Deleted*** = 1;  
 **public static final int *Blacked*** = 2;  
}

**public class** FriendRequest {  
 **public** String **reason**; //打招呼信息  
 **public** String **userID**;  
 **public** String **friendID**;  
 **public long timeStamp**;  
 **public int state**;  
 **public boolean isReaded**;  
}

**public class** GroupInfo {  
 **public** String **groupID**;  
 **public** String **name**;  
 **public** String **privateKey**;  
 **public** String **owner**;  
 **public** String **portrait**;  
 **public** List<MemberData> **userList**; //成员列表

}

**public class** MemberData {  
 **public** String **osnID**;  
 **public** String **groupID**;  
 **public int state**; //0-普通，1-所有者，2-管理员  
}

**public class** MessageInfo {  
 **public** String **userID**;  
 **public** String **target**;  
 **public** String **content**;  
 **public** String **timeStamp**;  
 **public** String **msgType**; *//text, voice, image, file, video, location, imageText, pText* **public boolean isGroup**;  
 **public** String **originalUser**;  
}

**public class** UserInfo {  
 **public** String **userID**;  
 **public** String **name**;  
 **public** String **displayName**;  
 **public** String **portrait**;  
}