**Rest API PHP版本调用说明**

Version 1.0

杭州叁体网络科技有限公司

2015/05/18

目录

1关键概念 3

1.1关键概念描述 3

1.2 角色 4

2．Jar包引用说明 4

2.1 jar包引入 4

2.1.1 avd-x.x.x.jar 4

2.1.2 json.jar 4

2.2 import jar包 5

3．接口说明 5

2.3 AuthServer 5

2.3.1 getAccessTocken() 5

2.3.2 AuthException 6

2.4 RoomServer 6

2.4.1 createRoom() 6

2.4.2 getRoomURL() 7

2.4.3 RoomException 7

**Rest API HPP版本调用说明**

杭州叁体网络科技有限公司是国内最专业的实时协同PaaS服务商，专注于音视频及数据实时协同领域，开发OpenAVD（Open Audio, Video, Data）中间件，支持html5兼容浏览器、Android/iOS、Windows/Mac等所有流行平台，以API/SDK的方式提供实时协同能力，方便第三方系统快速开发基于实时协同能力的各种App与应用。

本文详细描述Rest API PHP版本的调用说明，目标阅读者是第三方公司程序员。

## 1关键概念

## 1.1关键概念描述

|  |  |
| --- | --- |
| 关键概念 | 描述 |
| ACCESS\_KEY | 客户应用密钥, 会在传输中包含; |
| SECRET\_KEY | 客户安全密钥, 以SECRET\_KEY为参数，配合适当的签名算法，可以得到原始信息的数字签名，防止内容在传递过程中被伪造或篡改。客户必须保管好,不在网络上传输以防止被窃取 |
| ACCESS\_TOKEN | 管理凭证，是Open AVD用于验证管理请求合法性的机制。建议仅在业务服务器端使用这一类凭证，避免意外授权导致滥用。 |
| Room | 房间，是人与人交互的基本媒介，两人或多人之间的文字或音视频沟通，都在同一个Room中进行；参与者加入Room后，就可以和Room里其他参与者进行沟通与协同； |
| ROOM\_URL | 带房间标识的ＵＲＬ，可以微信、ＱＱ、邮件等发送给好友，让他人直接通过该ＵＲＬ加入房间进行沟通与协同。 |

## 1.2 角色

系统里定义了以下基本角色：

|  |  |
| --- | --- |
| 角色 | 描述 |
| 主持人 | * 每一个Room，有且只有一个主持人 * 主持人有若干控制能力，如打开关闭其他参与者的音视频，或把某些人请出Room * 主持人可以改变其他人的角色，如把参与者变成主持人等等 * 主持人有共享桌面或桌面应用程序的功能 |
| 主讲人 | * 主持人可指定Room中若干人成为主讲人 * 主讲人能够共享自己的屏幕或文档 * 没有控制功能 |
| 参与者 | * 能使用文字聊天，音视频等功能 * 没有控制功能 * 不能共享自己的本地桌面或文档 * 能接收显示主持人共享桌面的功能 |

## 2．PHP 文件引用说明

## PHP包引入

## rest.php

3tee公司提供的相关认证及创建会议等功能的php文件。

## include

include 'rest.php';

## 3．接口说明

以下详细描述各对象暴露出来的接口。

## Rest

### Rest ()

### 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | Rest () |
| 功能描述 | 构建方 |
| 调用说明 |  |
| 调用例子 | $restServer = "192.168.2.2";  $rest = new Rest($restServer); |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |
| restServer | 字符串 | Rest server，如"180.153.53.1:8081" |  |

### getAccessTocken()

* 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | getAccessTocken() |
| 功能描述 | 获取ACCESS\_TOKEN |
| 调用说明 |  |
| 调用例子 | $restServer = "192.168.2.2";  $accessKey = "demo\_access";  $secretKey = "demo\_secret";    try{  $rest = new Rest($restServer);  $accessTocken = $rest->getAccessTocken($accessKey,$secretKey);  echo "accessTocken = $accessTocken<br><br>";  }catch(AuthException $e){  //echo "$e";  $errorMessage = $e->errorMessage();  echo "AuthException:errorMessage.code =$errorMessage->code\n";  echo "errorMessage.message =$errorMessage->message<br><br>";  } |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |
| accessKey | 字符串 | 网络注册,获取的ACCESS\_KEY |  |
| secretKey | 字符串 | 网络注册,获取的SECRET\_KEY |  |

* 返回值

|  |  |  |  |
| --- | --- | --- | --- |
| 返回值 | 类型 | 描述 | 备注 |
| accessTocken | 字符串 | ACCESS\_TOKEN | 创建会议用到 |

### AuthException

异常罗列：

|  |  |  |
| --- | --- | --- |
| Code | Message | 备注 |
| 401 | un authored |  |
|  |  |  |

## 

### createRoom()

* 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | createRoom**()** |
| 功能描述 | 创建房间，获取房间ＩＤ |
| 调用说明 |  |
| 调用例子 | $roomType = 1  $topic = "toptest";  $maxAttendee= 50;  $maxVideo = 5;  $maxAudio = 5;  $hostPassword = "20";  $startTime = time();  $endTime = time()+60\*60\*3;  try{  $roomId = $rest->createRoom($accessTocken,$topic,$maxAttendee,$maxVideo, $maxAudio,$hostPassword,$startTime,$endTime,$roomType);  echo "roomId = $roomId<br><br>";    $room = $rest->getRoom($roomId);  echo "room.roomId = $room->roomId<br><br>";  echo "room.roomURL = $room->roomURL<br><br>";    $topic1 = "toptest111111";  $maxAttendee1= 30;  $maxVideo1 = 10;  $maxAudio1 = 20;  $updateRoomRet = $rest->updateRoom($roomId,$accessTocken,$topic1,$maxAttendee1,$maxVideo1, $maxAudio1,null,null,null,null);  echo "updateRoom, ret = $updateRoomRet<br><br>";    $ret = $rest->delRoom($accessTocken,$roomId);  echo "delRoom, ret = $ret<br><br>";    }catch(RoomException $e){  //echo "$e";  $errorMessage = $e->errorMessage();  echo "AuthException:errorMessage.code =$errorMessage->code\n";  echo "errorMessage.message =$errorMessage->message<br><br>";  } |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |
| accessTocken | 字符串 | ACCESS\_TOKEN |  |
| topic | 字符串 | 会议主题 |  |
| maxAttendee | Int | 允许最大参会人数 |  |
| maxVideo | Int | 允许最大视频路数 |  |
| maxAudio | Int | 允许最大音频路数 |  |
| hostPassword | 字符串 | 主持人密码，以该密码加会的，视为主持人 |  |
| startTime | Int | 会议开始时间，要求是GMT时间戳 | 非必填，可以设置为null |
| endTime | Int | 会议结束时间，要求是GMT时间戳 | 非必填，可以设置为null |
| roomType | Int | 会议类型，  1：临时会议，默认值;  说明：服务器根据会议 时间，会议在线人数，延迟时间后自动关闭会议;  2: 永久会议；  说明：主持人主动关闭会议才关闭，其他情况下一直保留会议 |  |

* 返回值

|  |  |  |  |
| --- | --- | --- | --- |
| 返回值 | 类型 | 描述 | 备注 |
| roomId | 字符串 | 房间ＩＤ |  |

### updateRoom()

1. 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | **updateRoom()** |
| 功能描述 | 更新房间信息 |
| 调用说明 |  |
| 调用例子 | 见2.6.1 |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |
| roomId | 字符串 | 会议号 |  |
| accessTocken | 字符串 | ACCESS\_TOKEN |  |
| topic | 字符串 | 会议主题 |  |
| maxAttendee | Int | 允许最大参会人数 |  |
| maxVideo | Int | 允许最大视频路数 |  |
| maxAudio | Int | 允许最大音频路数 |  |
| hostPassword | 字符串 | 主持人密码，以该密码加会的，视为主持人 |  |
| startTime | Int | 会议开始时间，要求是GMT时间戳 | 非必填，可以设置为null |
| endTime | Int | 会议结束时间，要求是GMT时间戳 | 非必填，可以设置为null |
| roomType | Int | 会议类型，  1：临时会议，默认值;  说明：服务器根据会议 时间，会议在线人数，延迟时间后自动关闭会议;  2: 永久会议；  说明：主持人主动关闭会议才关闭，其他情况下一直保留会议 |  |

* 返回值

|  |  |  |  |
| --- | --- | --- | --- |
| 返回值 | 类型 | 描述 | 备注 |
| ret | Int | 0： 代表更新成功 |  |

### getRoom()

* 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | **getRoom()** |
| 功能描述 | 获取房间对象，包括房间ＵＲＬ |
| 调用说明 |  |
| 调用例子 | 见2.6.1 |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |

* 返回值

|  |  |  |  |
| --- | --- | --- | --- |
| 返回值 | 类型 | 描述 | 备注 |
| room | Room | 房间对象 |  |

### delRoom()

* 接口基本信息

|  |  |
| --- | --- |
| 方法签名 | **delRoom()** |
| 功能描述 | 删除房间 |
| 调用说明 |  |
| 调用例子 | 见2.6.1 |

* 参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 描述 | 备注 |
| accessTocken | 字符串 | ACCESS\_TOKEN |  |
| roomId | 字符串 | 房间ＩＤ |  |

* 返回值

|  |  |  |  |
| --- | --- | --- | --- |
| 返回值 | 类型 | 描述 | 备注 |
| ret | Int | 删除状态，成功返回0 |  |

### RoomException

异常罗列：

|  |  |  |
| --- | --- | --- |
| Code | Message | 备注 |
| 401 | un un created |  |
|  |  |  |