



Agora Whiteboard SDK

参考手册

1.0 版

support@agora.io

目录

简介	3
Agora 实时云	3
Agora Whiteboard SDK	3
用户须知	4
兼容性	4
网页浏览器支持	4
文件格式支持	4
所需文档	5
已知问题和局限性	5
入门指南	6
获取 SDK	6
设置密钥方案	6
密钥介绍	6
获取和使用 Vendor Key	8
获取和使用 Dynamic Key	8
实现白板功能-网页应用程序	11
部署网页应用程序	11
运行网页示例程序	11
实现白板功能-非网页应用程序	13
Agora Whiteboard SDK – API 参考	14
join(params)	14
leave(url)	15

简介

Agora 实时云

Agora 实时云通过虚拟的 Agora Global Network 为客户提供基于互联网的音视频通信，在实时通信和移动对移动通信方面进行特别优化，确保满意的用户体验。Agora 实时云致力于解决移动设备的用户体验问题、3G/4G/Wi-Fi 网络性能各异、全球网络瓶颈等一系列问题，为用户带来优质的通信体验。

Agora 实时云包含以下 SDK，且 Native SDK 在程序生成时直接与应用程序建立连接：

- **Agora Native SDK for iOS 和 Agora Native SDK for Android**

用于移动设备的 Agora Native SDK，专为智能手机研发，其通信基于 Agora Global Network，并针对不同移动设备平台进行优化。

- **Agora Native SDK for Windows 和 Agora Native SDK for Mac**

- **Agora Native SDK for Web**

针对网页通信开发，为网页浏览器提供 Agora Global Network 通信支持。

- **Agora Whiteboard SDK**

Agora Whiteboard SDK 是 Agora 实时云 新增的功能，为异地用户之间的合作提供了一个简便的平台。用户可以在白板上直观地画图，注释以及分享 PDF 文档，方便沟通。

在下文的[入门指南](#)一节中将为您阐述如何使用 Agora Whiteboard SDK 实现白板功能。

Agora Whiteboard SDK

Agora Whiteboard SDK 是由 HTML 网页加载的一个 JavaScript 库。它还提供一组简单、高级的 JavaScript 应用程序接口，以便用户在 Agora Global Network 上建立白板通信。

Agora Whiteboard SDK 可采用 JavaScript 编程实现以下功能：

- **建立会话：**加入或退出 Agora 会话（作为标识，每个会话都有唯一的频道名称），可实现全球用户多方会话。
- **设置用户角色：**设置用户在会话里的角色 – 组织者或参与者。组织者可以在白板上画图、注释，和上传文件，而参与者仅有查看权限。

应用程序开发者可以通过以下两种方式访问 Agora 白板服务：

- 网页应用程序开发者可以利用 Agora Whiteboard SDK 提供的 JavaScript 类 [Agora.Whiteboard](#) 加入或退出会话，详见[实现白板功能-网页应用程序](#)和 [Agora Whiteboard SDK – API 参考](#)。
- 非网页应用程序开发者(在 iOS, Android, Mac 或 Windows 平台上)，可以通过控件 WebView 访问 Agora Whiteboard URL 服务，详见[实现白板功能-非网页应用程序](#)。

用户须知

在使用 Agora Whiteboard SDK 之前，请知悉以下内容。

兼容性

Agora Whiteboard SDK 与各平台 SDK 发行版本兼容。

网页浏览器支持

Agora Whiteboard SDK 需要与 HTML5 兼容的网页浏览器，下表列出了所有测试验证过的浏览器类型和版本，可在不同操作平台上运行：

	IE v10-11	Chrome v49-51	Safari v9.1	Firefox v46-47	Opera v38
HTTP/ HTTPS	Y	Y	Y	Y	Y

注：所有以上列出的浏览器均已经过测试和验证。之前的版本有可能也支持，但并未经过 Agora.io 测试验证。

文件格式支持

您可以上传和分享以下格式的 PDF 文件、微软 Office 文件以及图片：

文件	格式
PDF 文件	<ul style="list-style-type: none">• .pdf
Excel 文件	<ul style="list-style-type: none">• .xls• .xlsx
Word 文件	<ul style="list-style-type: none">• .doc• .docx
PowerPoint 文件	<ul style="list-style-type: none">• .ppt• .pptx
图片文件	<ul style="list-style-type: none">• .png• .jpg• .jpeg

注：上传的图片文件不支持 JPEG2000 格式。

所需文档

Agora Whiteboard SDK 仅实现白板功能，如想实现其他平台通话功能，详见：

- 各平台参考手册

各平台 SDK 以及参考手册，可在以下网址下载：

<http://cn.agora.io/download/>

已知问题和局限性

- 一个会话最多只能有一个白板实例。
- 可以在所有 Agora 支持的平台上运行，包括网页端，Mac 和 Windows 平台客户端。但在 iOS 和 Android 客户端仅能查看。
- 如果您的应用程序在 Mac 平台上，则不支持文件上传功能。该问题是由于 OS X 无法打开 WKWebView 操作面板、无法弹出文件上传窗口所致。须待 OS X 的更新。

入门指南

获取 SDK

请登录 <http://cn.agora.io/download/> 网站下载 Agora Native SDK for Android，或联系 sales@agora.io 获取最新版本。

组件	描述
./doc	Agora Whiteboard SDK 文档(中英文): <ul style="list-style-type: none">Agora_Whiteboard_SDK_Reference_Manual_v1_0_CHS.pdfAgora_Whiteboard_SDK_Reference_Manual_v1_0_EN.pdf
./client	Agora Whiteboard JavaScript 库和 web 版示例程序
./server	生成动态密钥所需的 Web 版服务器端示例代码

设置密钥方案

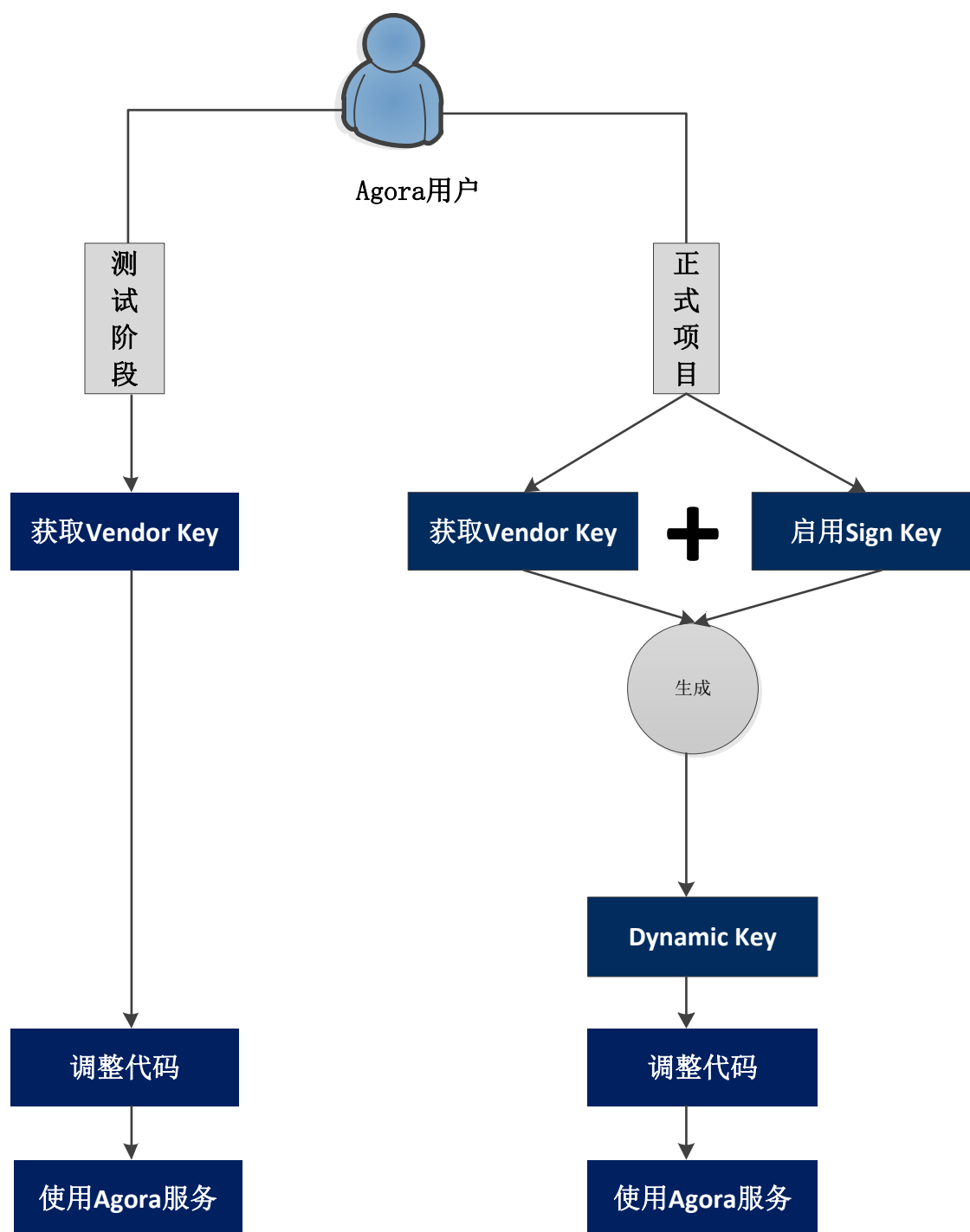
您在使用 Agora SDK 时可能需要用到多种密钥，本章主要介绍 Vendor Key, Sign Key 和 Dynamic Key 的概念和用法。

密钥介绍

每个在 Agora 官网注册的账户均能创建多个项目，且每个项目均有唯一的 Vendor Key。在 Agora Global Network 上的通信按照 Vendor Key 隔离，持有不同 Vendor Key 的用户即使加入的频道名称相同也会隔离，不会互相干扰。但 Vendor Key 是一种静态密钥，如果有人非法获取了您的 Vendor Key，他将可以在 Agora 提供的 SDK 中使用您的 Vendor Key，如果他知道您的频道名字，甚至有可能干扰您正常的通话。

Dynamic Key 提供更为安全的用户身份验证方案，用户每次访问 Agora 的服务时(例如加入会话时)，后台服务通过 Agora 提供的 Vendor Key 和 Sign Key, 基于 HMAC 的安全算法生成一个新的动态密钥发送给客户端，客户端在调用 Agora.Whiteboard.join(针对网页应用程序)或通过 WebView 控件访问 Agora Whiteboard URL(针对非网页应用程序)时使用此动态密钥。Agora 的服务器通过动态密钥验证用户是否合法。

注：Agora Whiteboard SDK 暂不支持录制功能。



注:

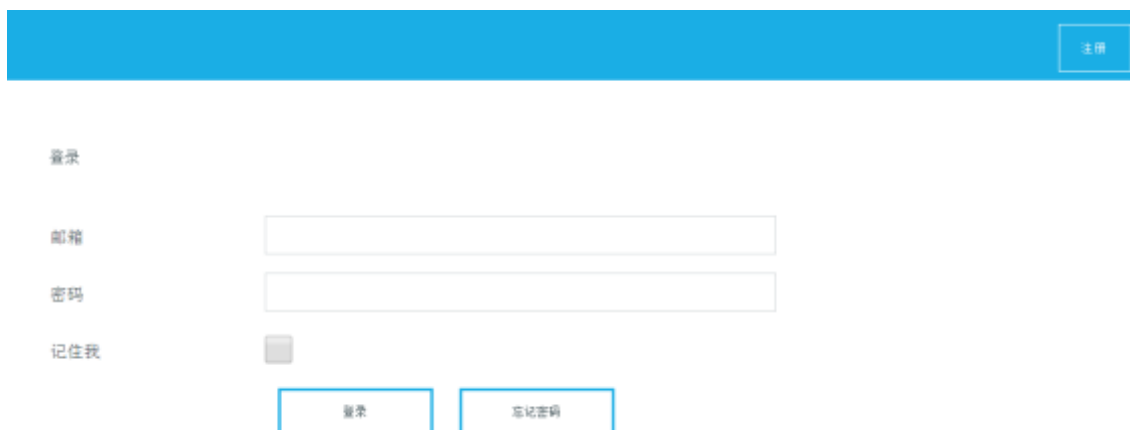
- 图为Agora建议的用法，测试阶段仅使用Vendor Key可以方便您快速搭建项目，正式项目建议使用Dynamic Key提高安全性。
- Sign Key仅用来生成Dynamic Key，不可单独使用。

获取和使用 Vendor Key

获取 Vendor Key

每个 Agora 账户下可创建多个项目，且每个项目有独立的 Vendor Key。

1. 访问 <https://dashboard.agora.io/>，点击右上角的**注册**，如下图所示：

The image shows a registration form on the Agora dashboard. At the top right, there is a blue button labeled '注册' (Register). Below this, the form includes a '登录' (Login) link, a '邮箱' (Email) input field, a '密码' (Password) input field, and a '记住我' (Remember me) checkbox. At the bottom of the form, there are two buttons: '登录' (Login) and '忘记密码' (Forgot password).

2. 在**项目列表**页面点击**新增**。
3. 填写**项目名**，如不需使用动态密钥，则不需选择**启用 Sign Key**。点击**保存**。
4. 在所创建的项目下查看 **Vendor Key**。获取 **Vendor Key** 后即可使用对应的 Agora.io 服务功能。

使用 Vendor Key

使用每个项目下唯一的 Vendor Key 访问对应的 Agora.io 服务功能：

1. 在示例程序的网页中输入 **Vendor Key**(仅针对网页应用程序)。
2. 加入会话时，将参数 **key** 设为 Vendor Key。

获取和使用 Dynamic Key

获取 Vendor Key 和 Sign Key

每个 Agora 账户下可创建多个项目，且每个项目有独立的 Vendor Key 和 Sign Key。

1. 登录 <https://dashboard.agora.io/>。
 2. 在**项目列表**页面点击**新增**。
 3. 填写**项目名**，选择**启用 Sign Key**。点击**保存**。
- 项目添加成功。



4. 查看已添加项目的 Vendor Key 和 Sign Key。

点击锁标志显示 Sign Key，再次点击该标志隐藏 Sign Key。

如果您在创建项目时没有选择**启用 Sign Key**，在项目添加之后，可以点击图中所示的**说明**来获取 Sign Key。

Sign Key: Sign Key尚未使用

说明

注:

- 如果出于某种原因您需要更新 Sign Key，请联系 support@agora.io。
- 将您的 Sign Key 保存在服务器端，且对任何客户端均不可见。
- Sign Key 用法，详见[表 1](#)。

集成 Dynamic Key 算法

开发者将生成动态密钥的算法集成到自己企业的信令服务中。Agora 提供了示例代码，涵盖 C++, Java, Python, node.js 等语言，可以直接将代码应用在您的程序中。

请登录以下网址获取示例代码：

- Github: <https://github.com/AgoraLab/AgoraDynamicKey>

使用 Dynamic Key

每一次客户端请求服务时：

1. 客户端请求企业自己的服务器的信令服务授权。
2. 服务器端基于 Sign Key、Vendor Key、频道名称、当前时间戳，客户端用户 id，有效期时间戳等信息通过 Agora 提供的算法生成动态密钥，返回给授权的客户端应用程序。
3. 客户端应用程序加入会话时，需将参数 key 设为 Dynamic Key。
4. Agora 的服务器接收到动态密钥信息，验证该会话是来自于合法用户，并允许访问 Agora Global Network。

动态密钥的安全性

采用动态密钥方案的企业，在用户加入会话时都需要提供动态密钥，确保用户每次会话都经过信令服务器的验证。动态密钥采用 HMAC/SHA1 签名方案，提高了系统及通话的安全性。

动态密钥结构

下表中所有字段从前往后拼接，共 103 字节。

字段	类型	长度	说明
版本号	字符串	3	动态密钥算法的版本信息
签名	字符串	40	<p>签名的 hex 编码。将 Sign Key 以及下列字段作为输入，通过 HMAC 计算和 hex 编码而成的 40 字节字符串：</p> <ul style="list-style-type: none"> ✓ 服务类型：ASCII 可见字符串，由 Agora 提供。详见章节 服务类型。 ✓ Vendor key: 32 位 vendor key 字符串。 ✓ Timestamp: 动态密钥生成时的时间戳。 ✓ 随机数: 32 位整数 hex 编码。随每个请求重新生成。 ✓ Channel: 频道名称，用户自定义，不超过 64 字节。 ✓ 用户 id: 客户端自定义的用户 id ✓ 服务到期时间戳: 用户在频道内截止通话的时间戳
Vendor Key	字符串	32	Vendor Key
授权时间戳	数字	10	动态密钥生成时的时间戳，自 1970.1.1 开始到当前时间的秒数。授权 5 分钟内可以访问 Agora 服务。
随机数	整数	8	32 位整数 hex 编码。随每个请求重新生成。

服务到期时间戳	数字	10	用户使用 Agora 服务终止的时间戳，在此时间之后，将不能继续使用 Agora 服务（例如进行的通话会被强制终止）；如果对终止时间没有限制，设置为 0。
---------	----	----	---

表 1

服务类型

服务	值	说明
通话服务	ACS	Agora 提供的白板会话服务，当用户加入会话时，需要传入动态密钥。

签名算法：HMAC/SHA1

动态密钥采用业界标准化的 HMAC/SHA1 加密方案，在 Node.js, Java, Python, C++ 等绝大多数通用的服务器端开发平台上均可获得所需库。具体加密方案可参看以下网页：

http://en.wikipedia.org/wiki/Hash-based_message_authentication_code

如需更多技术支持，请联系 Agora.io。

实现白板功能-网页应用程序

部署网页应用程序

运行 **Agora Whiteboard SDK** 的应用程序是标准的 JavaScript 程序，需加载 Agora JS 库并访问该 SDK 所提供的 JS 扩展库。

部署 JavaScript 库的方法跟一般的 JavaScript 主机程序相同：

1. 加载 Agora JS 库：

AgoraWBSDK-1.0.0.js

2. 需要以下 JS 扩展库：

vendor-bundle.js

运行网页示例程序

客户端

网页示例程序只需要静态的 **vendor key**（在示例程序网页上输入即可）。按照以下步骤运行示例程序：

1. 确保已安装本地网页服务器，如 Apache, NginX 或 Node.js。
2. 将 **./client/** 下的文件部署到网页服务器上，修改 **script-host.js** 和 **script-guest.js** (位于 **client/js** 文件夹下)，将 **Key** 值设置为用户的 **vendor key**，然后启动 http/https 服务。

3. 在网页服务器上用浏览器打开示例程序页面（关于推荐的浏览器，请参考章节 [网页浏览器支持](#)）。

如需使用**动态密钥**，请执行以下操作：

注：更多详细内容，请参考章节[获取和使用 Dynamic Key](#)。

1. 设置和启动生成动态密钥的服务器。
2. 从客户端向生成动态密钥的服务器发送 http/https 请求，获取动态密钥。
3. 在调用 `Agora.Whiteboard.join` 之前，将 `script-host.js` 和 `script-guest.js` 里 (位于文件夹 `client/js/`下)的 `Key` 替换成新生成的动态密钥。

服务器端

如需使用更为安全的**动态密钥**，可尝试以下代码。在实际使用时，开发者应将该方法应用到自己的服务器端程序上，并使用服务器上的现有的编程语言重新编程。本示例代码是 `JavaScript` 写成，因此需要标准的 `Node.js` 服务器上使用：

1. 在本地服务器或云主机上安装标准的 `Node.js` 服务器。
2. 在 `/server/nodejs/` 运行 `'npm install'`。
3. 在 `../server/nodejs/DemoServer.js` 下填写 `VENDOR_KEY` 和 `SIGN_KEY` 的值。
4. 用 `'node DemoServer.js'` 打开服务器。

实现白板功能-非网页应用程序

非网页应用程序开发者(在 iOS, Android, Mac 或 Windows 平台上), 可以通过控件 WebView 访问

Agora Whiteboard URL 服务:

- URL:

<https://wb.agora.io/html/agora-wb.html>

- 用法示例:

<https://wb.agora.io/html/agora-wb.html?key=key1&cname=name1&role=host&uinfo=user1>

参数	描述	必选或可选项?
key	Key可以任选其一: <ul style="list-style-type: none">- vendor key: 注册时由Agora系统提供- dynamic key: 由vendor key和sign key通过算法生成的密钥。有一台执行密钥的生成算法的NodeJS服务器。这是访问Agora Global Network推荐使用的最安全方法。	必选项
cname	为Agora白板会话提供唯一频道名称的字符串, 长度在64字节以内。 以下为支持的字符集范围 (共89个字符): a-z A-Z 0-9 空格 !#\$%& () + , - ; : < = . > ? @ [] ^ _ { } ~	必选项
role	用户角色: <ul style="list-style-type: none">● host 组织者● guest参与者 组织者可以控制白板, 参与者仅能查看	必选项
uinfo	用于识别用户的用户名, 如果不指定, 服务器将默认生成任意用户名。	非必选项
expire	白板频道存活时间(分钟数), 最高30天。 如果不指定, 默认在白板频道创建开始后24小时内有效。	非必选项
width	白板宽度, 默认为1024.	非必选项
height	白板高度, 默认为768.	非必选项

Agora Whiteboard SDK – API 参考

Agora Whiteboard SDK 库包含以下类:

Agora.Whiteboard	用 Agora.Whiteboard 对象加入和退出共享的会话
-------------------------	---------------------------------

join(params)

该方法让用户加入会话。如果用户是会话发起人，该方法将创建一个会话。

参数名称	类型	描述
params	对象	join 方法的唯一参数, 它包含下列设置。

key	字符串	Key可以任选其一: <ul style="list-style-type: none">- vendor key: 注册时由Agora系统提供- dynamic key: 由vendor key和sign key通过算法生成的密钥。 有一台执行密钥的生成算法的 NodeJS 服务器。这是访问 Agora Global Network 推荐使用的最安全方法。
cname	字符串	为Agora白板会话提供唯一频道名称的字符串, 长度在64字节以内。 以下为支持的字符集范围 (共89个字符): a-z A-Z 0-9 空格 !#\$%& (+,- :;<=. >? @[] ^_` { } ~
host	数值	用户角色: 1: 组织者 0: 参与者 组织者可以控制白板, 参与者仅能查看
expire	数值	(非必选项) 白板频道存活时间(分钟数), 最高30天 默认值为24小时
container	字符串	DOM 节点的身份。
width	数值	容器宽度

height	字数值	容器高度
uinfo	字符串	(非必选项) 用户名称

示例代码:

```
var hostParams = {
    key    : 'f4637604af81440596a54254d53ade29',
    cname  : 'PES-2017',
    host   : 1,
    width  : 800,
    height : 600,
    container : "wbHost"
};
/* Call AgoraWhiteBoardApi */
Agora.Whiteboard.join(hostParams);
```

leave(url)

该方法让用户退出会话。

参数名称	类型	描述
url	字符串	(非必选项) 指定用户将被重定向到的网址。.

示例代码:

```
Agora.Whiteboard.leave("http://sample.com");
```