

Server-SDK 对接 Demo 代码说明

版本：V 1.0.3

2017 年 7 月

文档修订记录

版本 编号	*变化 状态	变更内容和范围	变更日期	变更人	批准 日期	批准 人
1.0.0	A	添加 server-sdk 对接示例代码说明	2017/07/18	宋嘉		
1.0.1	M	修改获取授权信息 http 接口响应数据为授权用户所需数据	2017/07/20	宋嘉		
1.0.2	A	将 JavaScript-SDK 内容集成到 server-demo 中	2017/07/24	宋嘉		
1.0.3	M	创建通信通道接口返回的通道信息实体 ChannelInfo 中添加待生成二维码的原数据字符串 qrData 属性	2017/07/27	宋嘉		

*变化状态：A——增加，M——修改，D——删除，N——正式发布

目录

1	SERVER SDK 对接示例代码说明	4
1.1	SDK 调用说明	4
1.2	配置文件说明	4
1.3	演示页面访问	5
2	接口设计.....	5
2.1	实体定义	5
2.1.1	CreateChannelReq.....	5
2.1.2	GetAuthorizationInfoReq.....	5
2.1.3	ChannelInfo	6
2.1.4	AuthorizationInfoRes.....	6
2.2	接口定义	6
2.2.1	通信通道创建接口.....	6
2.2.2	授权信息获取接口.....	7
2.3	MAPPINGSERVER 获取数据相关逻辑说明	8
2.3.1	授权信息.....	8
3	附录	8
3.1	错误码	8

1 Server SDK 对接示例代码说明

- 此章节仅对 server 端调用 serv-sdk 的示例代码做一些介绍，具体的 server-sdk 接口明细请参考《数字身份 Server SDK 接口设计》文档说明；
- server-demo 中 http 的 API 接口设计，server-sdk 回调数据处理仅供参考，具体实现由第三方根据业务需求自行处理；
- 对接示例代码请参考《server-demo-src.zip》源码包；

1.1 SDK 调用说明

- SDK 调用及 server 端与 JavaScript-SDK 交互接口设计可参考示例代码中“com.vianet.open.demo.controllers.IMIController”类。
- 获取授权信息时调用 server-sdk 的 IMIAuthorizationRouter.getAuthorizationInfo() 接口需要传参 scope，此参数类型为字符串集合，存放需要获取的详细授权信息类型标识符，此类型定义与《第三方登录授权接口说明文档 V1.1》中的“4.3.1 二维码参数 scope 类型”章节定义一致：

授权作用域 scope	说明
snsapi_info	登录（当第三方需要数字身份用户的基本信息时，如：数字身份 id, 手机号等信息）
snsapi_idcard	授权（当第三方需要数字身份用户授权身份证信息时）

1.2 配置文件说明

- server-sdk 需要使用到两个配置文件：
1、通用参数配置文件：imi-config.properties

参数名称	描述	示例
通信地址		
web.mapping.getTopicId	MappingServer- 通信标识获取接口 URL	由我司提供
web.mapping.pullData	MappingServer- 数据获取接口 URL	
基本信息		
imi.mappingOpenid	合作伙伴标识	由我司分配
imi.vportId	数字身份号	现暂由我司分配，后期通过网络申请
imi.name	数字身份名称	由用户自定义
imi.ks.pass	私钥 KeyStore 密码	暂由我司提供，后期通过网络申请，用户自定义

- 2、区块链私钥存储 KeyStore 文件：imi-ks
KeyStore 文件暂由我司提供，后期通过网络申请；

- 配置文件存储位置：

1、应用程序内部存储：

存储于程序 classpath 根目录下，文件路径为

/META-INF/imi/imi-config.properties

/META-INF/imi/imi-ks

2、应用程序外部存储

存储于系统文件目录下，如 linux 下的存储位置如下：

/home/imi/imi-config.properties

/home/imi/imi-ks

在应用程序启动是通过调用 server-sdk 的参数初始化配置接口
“IMConfiguration.initConfigPath(String imiConfigPath, String imiKsPath)”传入两个配置文件绝对路径（参考示例代码中 StartBoot.initIMConfig() 中代码）。

1.3 演示页面访问

- 本演示示例代码中已经集成了 JavaScript-SDK 代码，可以直接运行本示例代码进行演示页面访问，**首页访问地址**：<http://host:port/index>
- 示例代码工程“src/main/resources/”目录下的“static”、“templates”目录内容即为原 JavaScript-SDK 相关内容。

2 接口设计

- 接口地址：<http://host:port/imi>
- 本章节所有请求数据格式为 JSON，请在请求 Header 中添加 **Content-Type** 为 **application/json; charset=utf-8**

2.1 实体定义

2.1.1 CreateChannelReq

名称	创建数据通道 Http 接口请求参数对象		CreateChannelReq
字段名	类型 (是否必填)	说明	备注
version	String(Y)	版本号	JS-SDK 定义
scope	String(Y)	授权作用域字符串	如：“snsapi_info, snsapi_idcard”

2.1.2 GetAuthorizationInfoReq

名称	获取授权信息 Http 接口请求参数对象		GetAuthorizationInfoReq
字段名	类型 (是否必填)	说明	备注
topicId	String(Y)	Mapping server 通信标示	
scope	String(Y)	授权作用域字符串	如：“snsapi_info, snsapi_idcard”

2.1.3 ChannelInfo

名称	MappingServer 通道信息 Http 接口响应对象		ChannelInfo
字段名	类型 (是否必填)	说明	备注
topicId	String(Y)	Mapping server 通信标示	作为生成二维码的数据
openId	String(Y)		
scope	String(Y)	授权作用域字符串	如： “snsapi_info, snsapi_idcard”
name	String(Y)	第三方数字身份名称	作为生成二维码的数据 此名称来源于配置文件 imi-config.properties 中的 “imi.name”参数
version	String(Y)	版本号	JS-SDK 定义
qrData	String(Y)	生成二维码的原数据串	页面上可以直接用此串来生成二维码

2.1.4 AuthorizationInfoRes

名称	授权信息 Http 接口响应对象		AuthorizationInfoRes
字段名	类型 (是否必填)	说明	备注
vportId	String(Y)	授权用户数字身份号	
cin	String(Y)	居民身份号	
name	String(Y)	居民姓名	
sex	String(Y)	居民性别	
authority	String(Y)	签发机关	
dateBirth	String(Y)	居民出生日期	
dateIssue	String(Y)	有效期开始日期	
dateExpiry	String(Y)	有效期结束日期	
userName	String(Y)	用户名称	
mobile	String(Y)	用户手机号	

2.2 接口定义

2.2.1 通信通道创建接口

接口名称	通信通道创建接口		
接入点	/createChannel	HTTP Method	POST
Header		签名	否
功能	获取 topicId 与第三方数字身份名称，作为生成二维码数据		

输入参数			输出参数 result		
version	String(Y)	版本号	result	ChannelInfo	通信标识信息
scope	String(Y)	授权作用域字符串			

示例：

请求	{ "version": "2.0", "scope": "snsapi_info, snsapi_idcard" }				
应答	{ "result": { "name": "DefaultName", "openId": "A8081676ED67F44DD9FC5E9347D1DC4B", "scope": "snsapi_idcard, snsapi_info", "topicId": "7c5cf577934e4c14bc81a9e4e573elfd", "version": "2.0", "qrData": "eyJmYXV1IjoiU2VydjVydVpVbWVyc2VvbiI6IjIuMCJ9", "retCode": "0000000", "retInfo": "success" }				

2.2.2 授权信息获取接口

接口名称	授权身份信息获取接口		
接入点	/getAuthorizationInfo	HTTP Method	POST
Header		签名	否
功能	此接口从 MappingServer 获取 app 推送的授权信息		
输入参数		输出参数 result	
topicId	String(Y)	MappingServer 通信标示	授权信息
scope	String(Y)	授权作用域字符串	

示例：

请求	{ "topicId": "7b698d0f6ac347939bf09b6299135b28", "scope": "snsapi_info, snsapi_idcard" }		
应答	无数据	{ "result": null, "retCode": "0000000", "retInfo": "success" }	
	授权身份数据	{ "result": { "authority": "北京", "cin": "172928199902031122", "dob": "", "doe": "2020-06-19", "doi": "2014-06-19", "mobile": "", "name": "李雷", "sex": "男", "userName": "", "vportId": "0x457b1573d81732ccaa8d8287279bfa48a915ecf1", "retCode": "0000000", "retInfo": "success" }	
	授权用户数据	{ "result": { "authority": "", "cin": "", "dob": "", "doe": "", "doi": "", "mobile": "13683238888", "name": "", "sex": "", "userName": "LiLei", "vportId": "0x457b1573d81732ccaa8d8287279bfa48a915ecf1", "retCode": "0000000", "retInfo": "success" }	
	授权所有数据	{ "result": { "authority": "北京", "cin": "172928199902031122", "dob": "", "doe": "2020-06-19", "doi": "2014-06-19", "mobile": "13683238888", "name": "李雷", "sex": "男", "userName": "LiLei", "vportId": "0x457b1573d81732ccaa8d8287279bfa48a915ecf1", "retCode": "0000000", "retInfo": "success" }	

2.3 MappingServer 获取数据相关逻辑说明

2.3.1 授权信息

1、首先通过 SDK 调用 IMIAuthorizationRouter.createChannel() 接口创建 MappingServer 通信通道，将返回的通道信息 ChannelInfo 响应给第三方页面（生成二维码）或第三方 app；

2、第三方页面或第三方 App 需要获取数据则通过 SDK 调用 IMIAuthorizationRouter.getAuthorizationInfo() 接口获取授权信息 AuthorizationInfo：

a、若 AuthorizationInfo 为 null 则说明从 mapping server 上未收到 app 推送的数据，正常返回，等待下一次获取；

b、若 AuthorizationInfo 非 null 则说明正常获取到 app 推送的授权信息，此时获取的授权人信息 Authorizer 一定是非 null 的，而授权登录用户信息 LoginUserInfo 与授权身份信息 IdentityCardInfo 就不一定非 null，根据接口调用时传的参数 scope 决定；

3、第三方用户 Http 接口响应信息 AuthorizationInfoRes 可根据业务需求自行从授权信息 AuthorizationInfo 对象中获取并组装。

3 附录

3.1 错误码

方法	说明
0000000	success
0000001	系统错误
0000002	参数错误