AHCA手机盾接口说明

安徽省电子认证管理中心有限责任公司

1. **工程配置**

以下内容以Android Studio开发工具为例。

1. 将手机盾SDK目录下后缀名为aar的文件拷贝到项目工程的libs目录下。
2. 如果您的项目中使用proguard等工具做了代码混淆，请保留以下选项（如果您的项目没有选择手机盾活体检测功能，仅保留第一条即可）：

**-keep class com.ahca.sts.\*\*{\*;}**

**-keep class cn.unitid.liveness.\*\*{\*;}**

**-keep class com.baidu.\*\*{\*;}**

点击Sync Project with Gradle Files按钮，同步工程配置。

1. **集成步骤**

初始化手机盾配置，建议在自定义Application类的onCreate生命方法内操作：

**STShield.getInstance()**

**.init(this, "服务地址", "AppKey", "SecretKey")**

**.initFace(this, "活体检测授权码", "活体检测授权文件名")**

**.initThemeColor(this, getResources().getColor(R.color.colorPrimary));**

**注意，同时需要在确定用户信息和唯一标识后调用以下接口：**

**STShield.getInstance()**

**.initUseId(this, "用户唯一标识");**

请根据实际需求进行配置，例如项目无需使用手机盾活体检测功能，则无需配置initFace。

在使用证书前（例如签名、签章等），需要先判断证书是否已申请，如果证书不存在，需要调用申请证书接口，申请成功后方可使用。建议使用以下几种实现方案：

1. 直接使用证书功能，例如调用证书签名/证书签章/证书登录等接口，如果得到返回码为10502，则再调用申请证书接口，申请成功后再次调用证书签名等对应接口进行操作。
2. 使用证书签名/签章/登录等接口前，先调用checkCert接口，检查证书状态，如果得到返回码为10502，则调用申请证书接口，申请成功后再次调用证书签名等对应接口进行操作，得到返回码为10503，则可以直接使用证书功能。
3. 调用isLocalCertExist接口，判断本地证书是否存在。

\*\*实际上A方案提到的所有涉及证书数据的接口，均包含了checkCert接口的功能，单独提供checkCert以及isLocalCertExist接口，为了方便多样化业务场景，例如，APP证书登录场景，由于checkCert、certLogin或其他接口需要联网，存在一定的耗时，可仅使用isLocalCertExist接口仅判断本地证书是否存在，证书存在则调用证书登录接口，证书不存在则显示密码输入框或验证码相关控件，进行后续申请证书的操作。

1. **应用系统接口**

## 配置服务地址、项目授权

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，配置SDK相关初始化参数，使用此接口 | | |
| 函数原型 | STShield init(Context context, String url, String appKey, String secretKey) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| url | String | 服务URL |
| appKey | String | 应用授权码 |
| secretKey | String | 应用授权码 |

## 配置用户标识

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，初始化用户唯一标识 | | |
| 函数原型 | STShield initUseId(Context context, String useId) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| useId | String | 用户唯一标识 |

## 配置活体检测授权

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，配置第三方活体检测的授权信息（如果业务无实名认证的需求，可以不调用该接口进行配置） | | |
| 函数原型 | STShield initFace(Context context, String faceLicenseID, String faceLicenseFileName) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| faceLicenseID | String | 授权文件的ID |
| faceLicenseFileName | String | 授权文件的FileName |

## 配置主题色

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，配置手机盾SDK主题颜色 | | |
| 函数原型 | STShield initThemeColor(Context context, int color) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| color | int | 需要使用getResources().getColor(R.color.white)或Color.WHITE格式 |

## 获取设备唯一标识

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，获取设备唯一标识，用于缓存PIN情况下的业务服务端直接调用手机盾服务进行签名等操作和签章的业务场景 | | |
| 函数原型 | String getDeviceID(Context context) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |

## 查询本地证书是否存在

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，判断证书申请状态（此接口只判断设备本地证书，如需实时比对服务端证书状态，建议使用checkCert接口） | | |
| 函数原型 | boolean isLocalCertExist(Context context) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |

## 检查证书状态是否正常

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，判断设备本地证书是否存在和判断本地证书与服务端证书状态是否正常 | | |
| 函数原型 | void checkCert(Context context, OnCheckCertListener result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| result | OnCheckCertListener |  |

## 申请个人证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void applyPersonalCert(Activity activity, StsUserInfo stsUserInfo, OnApplyCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| userInfo | StsUserInfo | 个人用户实体类 |
| result | OnApplyCertResult |  |

## 申请企业证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void applyCompanyCert(Activity activity, StsCompanyInfo stsCompanyInfo, OnApplyCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| companyInfo | StsCompanyInfo | 企业用户实体类 |
| result | OnApplyCertResult |  |

## 获取更换设备二维码

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务配置更换设备必须扫码跟换时，旧设备的业务端通过调用该接口，获取更换设备使用的二维码 | | |
| 函数原型 | void getUntieEquipmentQRcode(Context context, OnGetQRcodeResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| result | OnGetQRcodeResult |  |

## 更换设备

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务配置更换设备必须扫码跟换时，新设备的业务端扫描旧设备二维码后，通过调用该接口，申请证书并跟换证书绑定的设备 | | |
| 函数原型 | void untieEquipment(Activity activity, String qrData, OnApplyCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| qrData | String | 扫码得到的二维码信息 |
| result | OnApplyCertResult |  |

## 更新个人证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void updatePersonalCert(Context context, StsUserInfo stsUserInfo, OnUpdateCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| userInfo | StsUserInfo | 个人用户实体类 |
| result | OnUpdateCertResult |  |

## 更新企业证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void updateCompanyCert(Context context, StsCompanyInfo stsCompanyInfo, OnUpdateCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| companyInfo | StsCompanyInfo | 企业用户实体类 |
| result | OnUpdateCertResult |  |

## 重置个人证书PIN码

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void resetPersonalPIN(Context context, StsUserInfo stsUserInfo, OnResetPinResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| userInfo | StsUserInfo | 个人用户实体类 |
| result | OnResetPinResult |  |

## 重置企业证书PIN码

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void resetCompanyPIN(Context context, StsCompanyInfo stsCompanyInfo, OnResetPinResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| companyInfo | StsCompanyInfo | 企业用户实体类 |
| result | OnResetPinResult |  |

## 修改证书PIN码

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void modifyPIN(Context context, OnModifyPinResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| result | OnModifyPinResult |  |

## 证书登录

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certLogin(Activity activity, String data, String dataFormat, String dataType, String pn, OnCertLoginResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| data | String | 签名原文 |
| dataFormat | String | STShield.DATA\_FORMAT\_P1（P1）;  STShield.DATA\_FORMAT\_P7（P7）;  STShield.DATA\_FORMAT\_DEFAULT（默认）; |
| dataType | String | STShield.DATA\_TYPE\_ORIGINAL\_TO\_HEXADECIMAL（表示data是原文数据转成的16进制类型数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_HEXADECIMAL（表示data是16进制编码的byte类型数据）;  STShield.DATA\_TYPE\_CHINESE\_TO\_HEXADECIMAL（表示data是中文转成的16进制类型数据）;  STShield.DATA\_TYPE\_ORIGINAL（表示data是原文数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_BASE64（表示data是Base64编码的byte类型数据）; |
| pn | String | 项目编号（没有则传空值） |
| result | OnCertLoginResult |  |

## 证书签章

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certSeal(Activity activity, String pn, OnCertSealResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| pn | String | 项目编号（没有则传空值） |
| result | OnCertSealResult |  |

## 证书签名

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certSign(Activity activity, String data, String dataFormat, String dataType, String pn, OnCertSignResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| data | String | 签名原文 |
| dataFormat | String | STShield.DATA\_FORMAT\_P1（P1）;  STShield.DATA\_FORMAT\_P7（P7）;  STShield.DATA\_FORMAT\_DEFAULT（默认）; |
| dataType | String | STShield.DATA\_TYPE\_ORIGINAL\_TO\_HEXADECIMAL（表示data是原文数据转成的16进制类型数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_HEXADECIMAL（表示data是16进制编码的byte类型数据）;  STShield.DATA\_TYPE\_CHINESE\_TO\_HEXADECIMAL（表示data是中文转成的16进制类型数据）;  STShield.DATA\_TYPE\_ORIGINAL（表示data是原文数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_BASE64（表示data是Base64编码的byte类型数据）; |
| pn | String | 项目编号（没有则传空值） |
| result | OnCertSignResult |  |

## 证书验签

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certVerifySign(Context context, String data, String dataFormat, String signData, String dataType, OnCertVerifySignResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| data | String | 签名原文 |
| dataFormat | String | STShield.DATA\_FORMAT\_P1（P1）;  STShield.DATA\_FORMAT\_P7（P7）;  STShield.DATA\_FORMAT\_DEFAULT（默认）; |
| signData | String | 签名结果 |
| dataType | String | STShield.DATA\_TYPE\_ORIGINAL\_TO\_HEXADECIMAL（表示data是原文数据转成的16进制类型数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_HEXADECIMAL（表示data是16进制编码的byte类型数据）;  STShield.DATA\_TYPE\_CHINESE\_TO\_HEXADECIMAL（表示data是中文转成的16进制类型数据）;  STShield.DATA\_TYPE\_ORIGINAL（表示data是原文数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_BASE64（表示data是Base64编码的byte类型数据）; |
| result | OnCertVerifySignResult |  |

## 证书加密

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certEncrypt(Context context, String data, String dataFormat, String dataType, OnCertEncryptResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| data | String | 加密原文 |
| dataFormat | String | STShield.DATA\_FORMAT\_P1（P1）;  STShield.DATA\_FORMAT\_P7（P7）;  STShield.DATA\_FORMAT\_P7\_ENVELOPE（P7数字信封）;  STShield.DATA\_FORMAT\_DEFAULT（默认）; |
| dataType | String | STShield.DATA\_TYPE\_ORIGINAL\_TO\_HEXADECIMAL（表示data是原文数据转成的16进制类型数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_HEXADECIMAL（表示data是16进制编码的byte类型数据）;  STShield.DATA\_TYPE\_CHINESE\_TO\_HEXADECIMAL（表示data是中文转成的16进制类型数据）;  STShield.DATA\_TYPE\_ORIGINAL（表示data是原文数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_BASE64（表示data是Base64编码的byte类型数据）; |
| result | OnCertEncryptResult |  |

## 证书解密

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void certDecrypt(Activity activity, String decData, String dataFormat, String dataType, OnCertDecryptResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| decData | String | 加密结果 |
| dataFormat | String | STShield.DATA\_FORMAT\_P1（P1）;  STShield.DATA\_FORMAT\_P7（P7）;  STShield.DATA\_FORMAT\_P7\_ENVELOPE（P7数字信封）;  STShield.DATA\_FORMAT\_DEFAULT（默认）; |
| dataType | String | STShield.DATA\_TYPE\_ORIGINAL\_TO\_HEXADECIMAL（表示data是原文数据转成的16进制类型数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_HEXADECIMAL（表示data是16进制编码的byte类型数据）;  STShield.DATA\_TYPE\_CHINESE\_TO\_HEXADECIMAL（表示data是中文转成的16进制类型数据）;  STShield.DATA\_TYPE\_ORIGINAL（表示data是原文数据）;  STShield.DATA\_TYPE\_BYTE\_BY\_BASE64（表示data是Base64编码的byte类型数据）; |
| pn | String | 项目编号（没有则传空值） |
| result | OnCertDecryptResult |  |

## 变更证书状态

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void changeCertStatus(Context context, int statusType, OnChangeCertStatusResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| statusType | int | STShield.CERT\_STATUS\_TYPE\_FREEZE（冻结证书）;  STShield.CERT\_STATUS\_TYPE\_UNFREEZE（解冻证书）;  STShield.CERT\_STATUS\_TYPE\_REVOKE（废除证书）; |
| result | OnChangeCertStatusResult |  |

## 延期证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void postponeCert(Context context, OnPostponeCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| result | OnPostponeCertResult |  |

## 修改PIN缓存时间

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，可修改或清空服务器缓存PIN的时间 | | |
| 函数原型 | void setPrivateKeyCacheTime(Activity activity, String pn, OnPKCacheTimeResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| pn | String | 项目编号（没有则传空值） |
| result | OnPKCacheTimeResult |  |

## 清除PIN缓存时间

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，可清空服务器缓存PIN的时间 | | |
| 函数原型 | void clearPKCacheTime(Activity activity, String pn, OnPKCacheTimeResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| pn | String | 项目编号（没有则传空值） |
| result | OnPKCacheTimeResult |  |

## 保存签名图片（弹出签名面板）

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，弹出签名面板采集用户手写笔迹后保存至手机盾服务器，同时返回图片数据给业务端 | | |
| 函数原型 | void setSignImgWithDrawingBoard(Activity activity, SignImgSetting signImgSetting, OnSignImgResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| signImgSetting | SignImgSetting | 无自定义需求可传null或new SignImgSetting()  int userOrientation(面板方向)  {  STShield.ORIENTATION\_AUTO（根据重力感应自动切换）;  STShield.ORIENTATION\_LANDSCAPE（横屏）;  STShield.ORIENTATION\_PORTRAIT（竖屏）;  }  float paintStrokeWidth(画笔宽度)  int paintColor(画笔颜色)  int imgBackgroundColor(画板背景颜色)  int standardWidth(返回图片的宽度)  int standardHeight(返回图片的高度) |
| result | OnSignImgResult |  |

## 获取/保存签名图片（永久性数据）

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，可获取保存在手机盾服务器里的用户手写签名图片，如果服务器没有该用户的图片数据，则弹出签名面板采集用户手写笔迹后保存至服务器，同时返回图片数据给业务端 | | |
| 函数原型 | void getSignImgAndSetItIfNotExist(Context context, SignImgSetting signImgSetting, OnSignImgResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| signImgSetting | SignImgSetting | 无自定义需求可传null或new SignImgSetting()  int userOrientation(面板方向)  {  STShield.ORIENTATION\_AUTO（根据重力感应自动切换）;  STShield.ORIENTATION\_LANDSCAPE（横屏）;  STShield.ORIENTATION\_PORTRAIT（竖屏）;  }  float paintStrokeWidth(画笔宽度)  int paintColor(画笔颜色)  int imgBackgroundColor(画板背景颜色)  int standardWidth(返回图片的宽度)  int standardHeight(返回图片的高度) |
| result | OnSignImgResult |  |

## 获取签名图片（永久性数据）

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，可获取保存在手机盾服务器里的用户手写签名图片 | | |
| 函数原型 | void getSignImgFromService(Context context, OnSignImgResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| result | OnSignImgResult |  |

## 获取签名图片（一次性数据）

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 业务端通过调用该接口，弹出签名面板采集用户手写笔迹后返回图片数据给业务端 | | |
| 函数原型 | void getSignImgWithDrawingBoard(Context context, SignImgSetting signImgSetting, OnSignImgResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| signImgSetting | SignImgSetting | 无自定义需求可传null或new SignImgSetting()  int userOrientation(面板方向)  {  STShield.ORIENTATION\_AUTO（根据重力感应自动切换）;  STShield.ORIENTATION\_LANDSCAPE（横屏）;  STShield.ORIENTATION\_PORTRAIT（竖屏）;  }  float paintStrokeWidth(画笔宽度)  int paintColor(画笔颜色)  int imgBackgroundColor(画板背景颜色)  int standardWidth(返回图片的宽度)  int standardHeight(返回图片的高度) |
| result | OnSignImgResult |  |

## 获取证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void getCert(Context context, int certType, OnGetCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| certType | int | STShield.CERT\_TYPE\_SIGNCERT（签名证书）;  STShield.CERT\_TYPE\_ENCCERT（加密证书）; |
| result | OnGetCertResult |  |

## 获取单位编号列表

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void getDepartmentNo(Activity activity, OnGetDepartmentNoResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| result | OnGetDepartmentNoResult |  |

## 下载预制证书

|  |  |  |  |
| --- | --- | --- | --- |
| 函数原型 | void downloadCert(Activity activity, String phoneNum, String departmentNo, int certType, OnGetCertResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| activity | Activity |  |
| phoneNum | String | 手机号码 |
| departmentNo | String | 单位编号 |
| certType | int | STShield.CERT\_TYPE\_SIGNCERT（签名证书）;  STShield.CERT\_TYPE\_ENCCERT（加密证书）; |
| result | OnGetCertResult |  |

## 清理本地证书

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 用于清除设备本地证书数据 | | |
| 函数原型 | CommonResult clearCert(Context context) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |

## 查询生物识别启用状态

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 用于查询用户是否开启了使用生物识别代替PIN码进行验证 | | |
| 函数原型 | boolean getFingerprintStatus(Context context) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |

## 开启/关闭生物识别

|  |  |  |  |
| --- | --- | --- | --- |
| 功能描述 | 用于开启/关闭使用生物识别代替PIN码进行验证 | | |
| 函数原型 | void openFingerprint(Context context, boolean open, OnFingerprintResult result) | | |
| 参数 | 参数名 | 数据类型 | 备注 |
| context | Context |  |
| open | boolean | 开启(true) / 关闭(false) |
| result | OnFingerprintResult |  |

1. **实体类**

## ApplyCertResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| enCert | String | 加密证书 | Base64字符串 |
| signCert | String | 签名证书 | Base64字符串 |
| stsCertInfo | StsCertInfo | 证书实体类 |  |
| stsUserInfo | StsUserInfo | 个人用户信息 | 此信息只有业务配置需要做实名认证时才会返回 |
| stsCompanyInfo | StsCompanyInfo | 企业用户信息 | 此信息只有业务配置需要做企业认证时才会返回 |
| stsVHInfo | StsVHInfo | 虚拟人用户信息 | 此信息只有业务配置需要做虚拟人认证时才会返回 |

## CertLoginResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| signData | String | 签名结果 |  |
| signCert | String | 签名证书 |  |
| token | String | 缓存PIN后的token | 此信息只有业务配置了手机盾后台缓存PIN，并且用户同意后才会返回 |

## CertSealResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| signData | String | 签名结果 |  |
| signCert | String | 签名证书 |  |
| token | String | 缓存PIN后的token | 此信息只有业务配置了手机盾后台缓存PIN，并且用户同意后才会返回 |

## CertSignResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| signData | String | 签名结果 |  |
| signCert | String | 签名证书 |  |
| token | String | 缓存PIN后的token | 此信息只有业务配置了手机盾后台缓存PIN，并且用户同意后才会返回 |

## CertEncryptResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| encryptData | String | 加密结果 |  |

## CertDecryptResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| decryptData | String | 解密结果 |  |

## CommonResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |

## GetCertResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| enCert | String | 加密证书 | Base64字符串 |
| signCert | String | 签名证书 | Base64字符串 |
| stsCertInfo | StsCertInfo | 证书实体类 |  |

## GetQRcodeResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| qrCode | String | 二维码图片base64数据 |  |

## PKCacheResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| token | String | 缓存PIN的token |  |

## SignImgResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| token | String | 签章令牌 |  |
| signImg | byte[] | 签章图片 |  |

## GetDepartmentNoResult

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| resultCode | int | 返回码 |  |
| resultMsg | String | 返回信息 |  |
| departmentList | ArrayList<StsDepartment> | 单位编号列表 |  |

## StsDepartment

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| departmentNo | String | 单位编号 |  |
| departmentName | String | 单位名称 |  |

## StsCertInfo

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| publicKey | String | 公钥 |  |
| notBefore | String | 证书起始有效期 |  |
| notAfter | String | 证书失效日期 |  |
| certAlgorithm | String | 算法标识 |  |
| certSN | String | 签名证书序列号 |  |
| version | String | 证书版本号 |  |
| subjectDN | SubjectDNBean | 持有者主题 |  |
| issuerDN | IssuerDNBean | 发布者主题 |  |
| subjectEXT | SubjectEXTBean | 扩展项主题 |  |

## StsCertInfo.SubjectDNBean

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| ST | String | phoneNum |  |
| C | String |  |  |
| L | String | userCity |  |
| OU | String |  |  |
| O | String | idCardNum |  |
| CN | String | userName |  |
| E | String | userEmail |  |
| GIVENNAME | String | idCardNum |  |

## StsUserInfo

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| userName | String | 用户姓名 |  |
| cardType | String | 证件类型 | STShield.CARD\_TYPE\_ID\_CARD （身份证）;  STShield.CARD\_TYPE\_TEMPORARY\_ID\_CARD（临时身份证）;  STShield.CARD\_TYPE\_HOUSEHOLD\_REGISTER（户口本）;  STShield.CARD\_TYPE\_PASSPORT护照（护照）;  STShield.CARD\_TYPE\_MILITARY\_ID\_CARD（军人身份证）;  STShield.CARD\_TYPE\_ARMED\_POLICE\_ID\_CARD（武警身份证）; |
| cardNum | String | 证件号码 |  |
| phoneNum | String | 手机号码 |  |
| userCity | String | 城市 |  |
| userEmail | String | 邮箱 |  |
| certExt2 | String | 扩展项2 |  |
| certExt3 | String | 扩展项3 |  |
| certExt4 | String | 扩展项4 |  |

## StsCompanyInfo

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 属性类型 | 属性名称 | 属性说明 |
| companyName | String | 企业名称 |  |
| legalPerson | String | 法人姓名 |  |
| registerNo | String | 统一社会代码号 |  |
| cardNum | String | 法人身份证号码 |  |
| phoneNum | String | 法人手机号码 |  |
| userEmail | String | 公司邮箱 |  |
| certExt2 | String | 扩展项2 |  |
| certExt3 | String | 扩展项3 |  |
| certExt4 | String | 扩展项4 |  |

1. **错误码**

|  |  |  |
| --- | --- | --- |
| 错误码 | 标识名称 | 说明 |
| 1 | StsCodeTable.rtnCode\_success | 成功 |
| 10102 | StsCodeTable.rtnCode\_failure | 失败 |
| 10201 | StsCodeTable.rtnCode\_service | 服务异常 |
| 10202 | StsCodeTable.rtnCode\_network | 网络异常 |
| 10203 | StsCodeTable.rtnCode\_error | 服务返回参数为空 |
| 10204 | StsCodeTable.rtnCode\_service\_data\_empty | 服务返回数据为空 |
| 10302 | StsCodeTable.rtnCode\_pin\_error | 失败，PIN错误 |
| 10303 | StsCodeTable.rtnCode\_user\_canceled | 失败，用户点击取消 |
| 10304 | StsCodeTable.rtnCode\_repeated | 状态未更改，传入标识与之前重复 |
| 10305 | StsCodeTable.rtnCode\_fingerprint\_init\_failed | 指纹异常 |
| 10306 | StsCodeTable.rtnCode\_canceled | 用户取消 |
| 10501 | StsCodeTable.rtnCode\_no\_data | 参数不全或为空 |
| 10502 | StsCodeTable.rtnCode\_cert\_not\_exist | 证书不存在或已损坏 |
| 10503 | StsCodeTable.rtnCode\_cert\_exist | 证书已存在 |
| 10504 | StsCodeTable.rtnCode\_user\_info\_incomplete | 用户信息不全 |
| 10505 | StsCodeTable.rtnCode\_sign\_img\_error | 签名图片生成失败 |
| 10506 | StsCodeTable.rtnCode\_reviewing | 请完成PIN码的重置 |
| 10507 | StsCodeTable.rtnCode\_parameter\_error | 参数有误 |
| 10508 | StsCodeTable.rtnCode\_url\_not\_init | 未初始化服务地址 |
| 10509 | StsCodeTable.rtnCode\_url\_error | 服务器地址格式有误 |
| 10510 | StsCodeTable.rtnCode\_sdk\_data\_error | SDK内部数据传输错误 |
| 10511 | StsCodeTable.rtnCode\_face\_not\_init\_error | 未初始化人脸识别参数 |
| 10512 | StsCodeTable.rtnCode\_qr\_code\_error | 二维码错误 |
| 10513 | StsCodeTable.rtnCode\_pn\_is\_empty | 项目编号为空 |
| 10514 | StsCodeTable.rtnCode\_user\_refused | 用户拒绝 |
| 10517 | StsCodeTable.rtnCode\_pin\_num\_error | PIN码格式错误 |