**北京江南天安科技有限公司**

**应急广播签名验证接口**

**调用说明**



北京江南天安科技有限公司

**[2017-02-27]**

目录

[1 .概要说明 4](#_Toc476149427)

[1.1 范围 4](#_Toc476149428)

[1.2 产品信息 4](#_Toc476149429)

[1.3 开发环境 4](#_Toc476149430)

[2 接口说明 5](#_Toc476149431)

[2.1 接口文件说明 5](#_Toc476149432)

[3 功能接口说明 5](#_Toc476149433)

[3.1 连接SJJ1313密码器设备 5](#_Toc476149434)

[3.2 连接SJJ1507密码机设备 5](#_Toc476149435)

[3.3 计算数据签名 6](#_Toc476149436)

[3.4 验证签名值 7](#_Toc476149437)

[3.5 计算数据签名 8](#_Toc476149438)

[3.6 验证签名值 9](#_Toc476149439)

[3.7 十六进制数据转换为字符串 9](#_Toc476149440)

[3.8 字符串转换为十六进制数据 9](#_Toc476149441)

[3.9 对数据进行BASE64编码 10](#_Toc476149442)

[3.10 对数据进行BASE64解码 10](#_Toc476149443)

[3.11 计算数据摘要 11](#_Toc476149444)

[3.12 关闭设备 12](#_Toc476149445)

[4.Windows和Linux平台下的使用说明 12](#_Toc476149446)

[4.1 SJJ1313密码器的使用说明 12](#_Toc476149447)

[4.2 SJJ1507密码机的使用说明 13](#_Toc476149448)

[5.附录 14](#_Toc476149449)

[5.1 SJJ1313错误码说明 14](#_Toc476149450)

[5.2 SJJ1507错误码说明 17](#_Toc476149451)

[5.3 应急广播命令错误码说明 21](#_Toc476149452)

# .概要说明

## 范围

此文档用于描述不同密码设备（包括SJJ1313密码器、SJJ1507密码机等）为应急广播业务定制的签名验证功能的功能接口的定义及使用说明。

## 产品信息

适用于SJJ1313密码器和SJJ1507密码机等。

## 开发环境

开发语言：JAVA

开发环境：Windows + MyEclipse + JDK 1.7

接口包括32位及64位环境；

# 接口说明

## 接口文件说明

Java接口包文件列表

|  |  |
| --- | --- |
| **文件名** | **描述** |
| YingJGBCALLDLL20170224.jar | 接口框架层定义jar包 |
| jna-4.2.2.jar | 第三方jar包 |

常用功能类路径列表

|  |  |
| --- | --- |
| **说明** | **类路径（包含类名）** |
| 上层业务功能类 | cn.tass.yingjgb.YingJGBCALLDLL |

# 功能接口说明

## 连接SJJ1313密码器设备

|  |  |  |
| --- | --- | --- |
| 原型 | public static void openDevice() throws Exception{}; | |
| 描述 | 连接SJJ1313密码器设备，无参数，无返回值，若是打开设备失败，则会抛出异常。 | |
| 异常 | Exception | F0000107：Socket连接错误，打开会话失败 |
| 备注 | 调用事例如下： YingJGBCALLDLL.openDevice();//打开SJJ1313密码器 | |

## 连接SJJ1507密码机设备

|  |  |  |
| --- | --- | --- |
| 原型 | public static void openDevice(int flag) throws Exception{}; | |
| 描述 | 连接SJJ1507设备，有参数，无返回值，若是打开设备失败，则会抛出异常。 | |
| 参数 | Flag | 一个打开SJJ1507密码机设备的标识，参数类型为整数即可。 |
| 异常 | Exception | 01000139: 读取配置文件失败,打开会话失败 |
| 备注 | 调用事例如下： YingJGBCALLDLL.openDevice(0); // 打开sjj1507密码机 | |

## 计算数据签名

|  |  |  |
| --- | --- | --- |
| 原型 | public static byte[] sign (int dataType ,byte[] inData) throws Exception{} | |
| 描述 | 计算数据的签名值，输入的待签名的数据类型为字节数组。 | |
| 参数 | dataType | 待签名的数据类型：  0:指令数据； 1:消息数据； |
| inData | 待计算签名的数据 |
| 返回值 | Byte[] | 签名值 |
| 异常 | Exception | the value of dataType must be 0 and 1； |
| 备注 | 注意：计算消息摘要时，用消息原文+记数器进行计算；计算指令摘要时，用记数器+指令原文进行计算，注意计数器所处的位置差异。  1.指令签名的输入内容：待签名数据；  指令签名的输出内容：签名记数器4字节+签名证书编号6字节+签名值(64字节)；  **byte**[] data = **new** **byte**[]{17,18,19,20};  //签名值  byte[] signature = YingJGBCALLDLL.sign(0,data);  以十六进制形式输出：DC4411E6D2D2B424E2895E15233B8E3FAAB3F43274DD88DC0FA2F69A2FE0AA7469F7A4D5E529719BF3D5437A382177AF39BA0E952F05F1C5CBAB688389E782C0  //签名记数器  byte[] signCounter = YingJGBCALLDLL.*signCounter*;  以十六进制形式输出：000520B0  //签名证书编号  byte[] signCerSN = YingJGBCALLDLL.*signCerSN*;  以十六进制形式输出：000000000085  2.消息签名的输入内容：待签名数据；  消息签名的输出内容：签名记数器4字节+签名证书编号6字节+签名值(64字节)；  **byte**[] data = **new** **byte**[]{17,18,19,20};  byte[] signature = YingJGBCALLDLL.sign(1,data);  以十六进制形式输出：数据签名D468DB668A97BE7EF2220D69A976F76A688285ECC7E9944C80C9EBF46C8A70A933715F7318D6A1318D465A67A930A0963C3263A956B32D22C533583105195535  byte[] signCounter = YingJGBCALLDLL.*signCounter*;  以十六进制形式输出：  000520B1  byte[] signCerSN = YingJGBCALLDLL.*signCerSN*;  以十六进制形式输出：  000000000085 | |

## 验证签名值

|  |  |  |
| --- | --- | --- |
| 原型 | public static boolean verifySign(int dataType ,byte[] inData,byte[] signature) throws Exception{} | |
| 描述 | 验证数据的签名值 | |
| 参数 | dataType | 待签名的数据类型  0:指令数据 ； 1:消息数据 ； |
| inData | 输入的待签名数据 |
| Signature | 签名值 |
| 返回值 | Boolean | 返回结果  True:验证成功  False:验证失败 |
| 异常 | Exception | verify the signature is failed,the error code is FFFFFFFF;验签失败 |
| 备注 | 注意：计算消息摘要时，用消息原文+记数器进行计算；计算指令摘要时，用记数器+指令原文进行计算，注意计数器所处的位置差异。  1.指令验签的输入内容：待验证原文（N字节）+签名记数器+签名证书编号+签名值(64字节)；  指令验签的输出内容：验签结果；  **byte**[] data = **new** **byte**[]{17,18,19,20};  //签名值  byte[] signature = YingJGBCALLDLL.sign(0,data);  //签名记数器  byte[] signCounter = YingJGBCALLDLL.*signCounter*;  //签名证书编号  byte[] signCerSN = YingJGBCALLDLL.*signCerSN*;  **boolean** verify = YingJGBCALLDLL.*verifySign*(0, data, signature);// 验证数据签名  verify = true;  2．消息验签的输入内容：待验证原文（N字节）+签名记数器+签名证书编号+签名值(64字节);  消息验签的输出内容：验签结果；  **byte**[] data = **new** **byte**[]{17,18,19,20};  byte[] signature = YingJGBCALLDLL.sign(1,data);  byte[] signCounter = YingJGBCALLDLL.*signCounter*;  byte[] signCerSN = YingJGBCALLDLL.*signCerSN*;  **byte**[] signature1 = YingJGBCALLDLL.*sign*(1, data);// 数据签名  **boolean** verify1 = YingJGBCALLDLL.*verifySign*(1, data, signature1);// 验证数据签名；  verify1 = true; | |

## 计算数据签名

|  |  |  |
| --- | --- | --- |
| 原型 | public static String sign(int dataType,String inData) throws Exception{} | |
| 描述 | 计算数据的签名值，输入的数据类型为字符串 | |
| 参数 | dataType | 待签名的数据类型  0:指令数据 ； 1:消息数据； |
| inData | 待计算签名的数据 |
| 返回值 | String | 返回String类型的签名值 |
| 异常 | Exception | the value of dataType must be 0 and 1； |
| 备注 | 注意：计算消息摘要时，用消息原文+记数器进行计算；计算指令摘要时，用记数器+指令原文进行计算，注意计数器所处的位置差异。  输入的数据格式为字符串，返回数据的签名结果,格式:[N字节签名数据原文 || 4字节记数器 || 6字节签名证书号 || 64字节签名值]；  String data = "12345678";  String signature= YingJGBCALLDLL.*sign*(0, data);  signature ：12345678000520B9000000000085769410928A7F78A6BE0E333FAEB4721589641CF4C8EBF5300D7C25E7CD78B6D9AEA5AD7F57B75383ABEC896EBF8536ADEC1BA03721FF32523836BE854C3FDD49  String data = "12345678";  String signature= YingJGBCALLDLL.*sign*(1, data);  signature ：12345678000520BC000000000085FE67E136A9D2D7EAF63F641DA935B7E7A53457FFA3BC1EAC98D3AFB926CD38D226C08B0E1566A10C5ED96C0B8399469DE4BB03FB81DD387B828E7075D11A5AA3 | |

## 验证签名值

|  |  |  |
| --- | --- | --- |
| 原型 | public static boolean verifySign(int dataType,String inData) throws Exception{} | |
| 描述 | 验证数据的签名值 | |
| 参数 | dataType | 待签名的数据类型  0:指令数据 ； 1:消息数据文件(XML等) ； |
| inData | 输入的签名值 |
| 返回值 | Boolean | 返回结果  True:验证成功  False:验证失败 |
| 异常 | Exception | verify the signature is failed ,the error code is FFFFFFFF;验签失败 |
| 备注 | String data = "12345678";  String signature= YingJGBCALLDLL.*sign*(0, data);  boolean flag = YingJGBCALLDLL.verifySign(0, signature);  flag = true;  String data = "12345678";  String signature1= YingJGBCALLDLL.*sign*(1, data);  boolean flag = YingJGBCALLDLL.verifySign(1, signature1);  flag = true; | |

## 十六进制数据转换为字符串

|  |  |  |
| --- | --- | --- |
| 原型 | public static String Hex2Ascii(byte[] hexdata) throws Exception{} | |
| 描述 | 十六进制数据转换为字符串 | |
| 参数 | Hexdata | 输入的十六进制数据 |
| 返回值 | String | 转换之后的字符串 |
| 异常 | Exception | the value of hexdata is null; |
| 备注 | **byte**[] hexdata = **new** **byte**[]{5,6,11,23,16,18,19,20,4,52,15,88};  String asc = YingJGBCALLDLL.*Hex2Ascii*(hexdata);  Asc: 05060B171012131404340F58 | |

## 字符串转换为十六进制数据

|  |  |  |
| --- | --- | --- |
| 原型 | public static byte[] Ascii2Hex(String inData) throws Exception{} | |
| 描述 | 字符串转换为16进制数据 | |
| 参数 | inData | 输入的字符串 |
| 返回值 | Byte[] | 返回16进制数据 |
| 异常 | Exception | the value of hexdata is null; |
| 备注 | **byte**[] hexdata = **new** **byte**[]{5,6,11,23,16,18,19,20,4,52,15,88};  String asc = YingJGBCALLDLL.*Hex2Ascii*(hexdata);  byte[] hex = YingJGBCALLDLL.Ascii2Hex(asc);  hex:05060B171012131404340F58 | |

## 对数据进行BASE64编码

|  |  |  |
| --- | --- | --- |
| 原型 | public static String Base64Encode(byte[] data) throws Exception{} | |
| 描述 | 字符串转换为16进制数 | |
| 参数 | data | 输入的数据 |
| 返回值 | String | 返回编码之后的字符串 |
| 异常 | Exception | the value of data is null; |
| 备注 | **byte**[] str ={1,2,3,4,5};  String eData = YingJGBCALLDLL.*Base64Encode*(str);  eData: AQIDBAU= | |

## 对数据进行BASE64解码

|  |  |  |
| --- | --- | --- |
| 原型 | public static byte[] Base64Decode(String edata) throws Exception{} | |
| 描述 | 进行BASE64解码 | |
| 参数 | edata | 输入BASE64编码之后的数据 |
| 返回值 | Byte[] | 返回解码之后的数据 |
| 异常 | Exception | the value of edata is null; |
| 备注 | String eData = “AQIDBAU”;  byte[] Ddata = YingJGBCALLDLL.Base64Decode(eData);  Ddata: [1, 2, 3, 4, 5] | |

## 计算数据摘要

|  |  |  |
| --- | --- | --- |
| 原型 | public static byte[] calcHash(int hashAlg,byte[] inData) throws Exception{} | |
| 描述 | 计算数据摘要值 | |
| 参数 | hashAlg | 调用SJJ1313时候支持的HASH算法标识：  1 ：SM3算法  2 ： SHA-1算法  4 ： SHA256杂凑算法  调用SJJ1507时候支持的HASH算法标识：  1: SM3杂凑算法 SM3-256  2: SHA\_1杂凑算法  4: SHA\_256杂凑算法  8: MD5杂凑算法  16: SHA\_224杂凑算法  32: SHA\_384杂凑算法  64:SHA\_512杂凑算法 |
| inData | 输入的数据 |
| 返回值 | Byte[] | 返回Hash值 |
| 异常 | Exception | calculate hash is failed ,the error code is 01000928;hash算法标识不支持 |
| 备注 | byte[] data = new byte[]{17,18,19,20};  byte[] hash = YingJGBCALLDLL.calcHash(4, data);  以十六进制输出：  Hash: 1AA2454E38E78FBE098BE157EE8112D85D6E667E2304FDE4220850FAA2F46341 | |

## 关闭设备

|  |  |  |
| --- | --- | --- |
| 原型 | public static void closeDevice() throws Exception{} | |
| 描述 | 关闭设备：无参数，无返回值，若是关闭设备失败，则会抛出异常。 | |
| 异常 | Exception | 接口抛出异常 |
| 备注 | ingJGBCALLDLL.closeDevice();// 关闭设备 | |

# 4.Windows和Linux平台下的使用说明

SJJ1313和SJJ1507分别使用于windows操作系统和linux操作系统，以下是使用的相关配置说明，首先解压压缩包。

## 4.1 SJJ1313密码器的使用说明

1、在Windows操作系统下并且jdk的位数为32的情况之下，需要将TassSJJ1313&SJJ1507\_20170224\ \libs\sjj1313\win32下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1313下面的cardcfg.ini配置文件共同放在

C:\Windows\windows\_x32该路径下；

2. 在Windows操作系统下并且jdk的位数为64的情况之下，需要将TassSJJ1313&SJJ1507\_20170224\ \libs\sjj1313\win64下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1313下面的cardcfg.ini配置文件共同放在

C:\Windows\windows\_x64该路径下。

3、在Linux操作系统下并且jdk的位数为32需要将TassSJJ1313&SJJ1507\_20170224\libs\sjj1313\linux32文件夹下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1313下面的cardcfg.ini配置文件以及TassSJJ1313&SJJ1507\_20170224\ jna-4.2.2.jar以及TassSJJ1313&SJJ1507\_20170224\ YingJGBCALLDLL20170223.jar包存放到/usr/lib路径下面；

4、在Linux操作系统下并且jdk的位数为64需要将TassSJJ1313&SJJ1507\_20170224\libs\sjj1313\linux64文件夹下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1313下面的cardcfg.ini配置文件以及TassSJJ1313&SJJ1507\_20170224\ jna-4.2.2.jar以及TassSJJ1313&SJJ1507\_20170224\ YingJGBCALLDLL20170223.jar包存放到/usr/lib路径下面；

## 4.2 SJJ1507密码机的使用说明

1、在Windows操作系统下并且jdk的位数为32的情况之下，需要将TassSJJ1313&SJJ1507\_20170224\ \libs\sjj1507\win32下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1507下面的tacipher.ini配置文件共同放在

C:\Windows\windows\_x32该路径下；

2. 在Windows操作系统下并且jdk的位数为64的情况之下，需要将TassSJJ1313&SJJ1507\_20170224\ \libs\sjj1507\win64下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1507下面的tacipher.ini配置文件共同放在

C:\Windows\windows\_x64该路径下。

3、在Linux操作系统下并且jdk的位数为32需要将TassSJJ1313&SJJ1507\_20170224\libs\sjj1507\linux32文件夹下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1507下面的tacipher.ini配置文件以及TassSJJ1313&SJJ1507\_20170224\ jna-4.2.2.jar以及TassSJJ1313&SJJ1507\_20170224\ YingJGBCALLDLL20170223.jar包存放到/usr/lib路径下面；

4、在Linux操作系统下并且jdk的位数为64需要将TassSJJ1313&SJJ1507\_20170224\libs\sjj1507\linux64文件夹下面的所有动态库以及TassSJJ1313&SJJ1507\_20170224\libs\sjj1507下面的tacipher.ini配置文件以及TassSJJ1313&SJJ1507\_20170224\ jna-4.2.2.jar以及TassSJJ1313&SJJ1507\_20170224\ YingJGBCALLDLL20170223.jar包存放到/usr/lib路径下面；

# 5.附录

SJJ1313密码器的错误码和SJJ1507的错误码都需要和应急广播指令错误码异或之后得到最终的错误码。

## 5.1 SJJ1313错误码说明

SDR\_OK 0x0 //操作成功

SDR\_BASE 0x01000000 //错误码基础值

SDR\_UNKNOWERR SDR\_BASE + 0x00000001 //未知错误

SDR\_CONFIGFILE SDR\_BASE + 0x00000002 //配置文件找不到或错误(未启用，下一版本启用)

SDR\_RESULT\_FORMAT SDR\_BASE + 0x00000003 //返回数据格式错误

SDR\_RESULT\_STATUS SDR\_BASE + 0x00000004 //返回数据状态字错误

SDR\_OPENDEVICE SDR\_BASE + 0x00000005 //打开设备失败(连不到服务)

SDR\_OPENSESSION SDR\_BASE + 0x00000006 //创建会话失败(连不到服务或服务拒绝接受)

SDR\_KEYNOTEXIST SDR\_BASE + 0x00000007 //不存在的密钥调用

SDR\_ALGNOTSUPPORT SDR\_BASE + 0x00000008 //不支持的算法调用

SDR\_ALGMODNOTSUPPORT SDR\_BASE + 0x00000009 //不支持的算法模式调用

SDR\_RESULT\_CHECK SDR\_BASE + 0x0000000A //返回数据校验失败

SDR\_SM2\_ENCDATAERR SDR\_BASE + 0x0000000E //SM2加密数据错误

SDR\_FILESIZEERR SDR\_BASE + 0x00000017 //文件长度超出限制

SDR\_FILENOEXIST SDR\_BASE + 0x00000018 //指定的文件不存在

SDR\_FILEOFSERR SDR\_BASE + 0x00000019 //文件起始位置错误

SDR\_KEYTYPEERR SDR\_BASE + 0x0000001E //密钥类型错误

SDR\_KEYERR SDR\_BASE + 0x0000001F //密钥错误

SDR\_HANDLE\_CLOSED SDR\_BASE + 0x00000020 //函数执行过程中，该会话已经关闭。

SDR\_HANDLE\_COUNT SDR\_BASE + 0x00000021 //打开的会话太多。

SDR\_HANDLE\_INVALID SDR\_BASE + 0x00000022 //指定的会话句柄无效。

SDR\_NOT\_LOGIN SDR\_BASE + 0x00000025 //私钥使用权限未获取。

SDR\_INPUT\_LEN\_ERROR SDR\_BASE + 0x00000026 //输入参数长度错误。

SDR\_KEYID\_INVALID SDR\_BASE + 0x00000027 //含义：指定的密钥号非法。

SDR\_MECHANISM\_INVALID SDR\_BASE + 0x00000028 //含义：机制无效。

SDR\_NOT\_INITIALIZED SDR\_BASE + 0x00000029 //含义：未调用初始化。

SDR\_ALREADY\_INITIALIZED SDR\_BASE + 0x0000002A //含义：初始化已调用。

SDR\_DEVICEHANDLE\_INVALID SDR\_BASE + 0x0000002B //设备句柄无效

SDR\_OUTPUT\_LEN\_ERROR SDR\_BASE + 0x0000002C //输出长度不正确

SDR\_OUTPUT\_BUFFER\_IVVALID SDR\_BASE + 0x0000002D //输出存储区无效

SDR\_KEY\_MEM\_FULL SDR\_BASE + 0x0000002E

SDR\_SESSION\_INIT SDR\_BASE + 0x00000031 // 创建会话失败

SDR\_KEYPAIR\_LENGTH SDR\_BASE + 0x00000032 // 非对称密钥的指数长度

SDR\_PADDING\_RSA SDR\_BASE + 0x00000033 // RSA padding 错误

SDR\_UNPADDING\_RSA SDR\_BASE + 0x00000034 // RSA unpadding 错误

SDR\_HANDLE\_SYMKEY\_INVALID SDR\_BASE + 0x00000035 // 会话密钥句柄无效

SDR\_PARAMETER\_ERR SDR\_BASE + 0x00000037 //接口参数错误或不能识别

ERROR\_COMMAND\_FORMAT 0xF0000001 //报文不能被解析(错误报文)

ERROR\_DEVICE\_OPEN 0xF0000002 //打开USB设备失败(密码器掉线或故障)

ERROR\_USB\_WRITE 0xF0000003 //USB设备写失败

ERROR\_USB\_READ 0xF0000004 //USB设备读失败

ERROR\_USB\_READ\_TIMEOUT 0xF0000040 //USB任务超时

ERROR\_LINK\_MAX 0xF0000005 //客户端连接数量超出上限

ERROR\_USB\_MALFUNCTION 0xF0000006 //USB设备故障

PCIAPIERR\_SOCK\_CONNECT 0xF0000007 //Socket连接服务错误

PCIAPIERR\_SOCK\_SELECT\_INSEND 0xF0000008 //socket select调用错误,在发送时

PCIAPIERR\_SOCK\_SELECT\_INRECV 0xF0000009 //socket select调用错误,在接收时

PCIAPIERR\_SOCK\_SEND 0xF000000A //socket send错误

PCIAPIERR\_SOCK\_RECV 0xF000000B //socket recv错误

PCIAPIERR\_SOCK\_INVALID 0xF000000C //socket无效

PCIAPIERR\_SOCK\_CLOSE 0xF000000D //socket被对方关闭

PCIAPIERR\_SOCK\_TIMEOUT\_SEND 0xF000000E //socket发送超时

PCIAPIERR\_SOCK\_TIMEOUT\_RECV 0xF000000F //接收超时

## 5.2 SJJ1507错误码说明

SDR\_OK 0x00000000 // 操作成功

SDR\_BASE 0x01000000 // 错误码基础值

SDR\_UNKNOWERR SDR\_BASE + 0x00000001 // 未知错误

SDR\_NOTSUPPORT SDR\_BASE + 0x00000002 // 不支持的接口调用

SDR\_COMMFAIL SDR\_BASE + 0x00000003 // 与设备通信失败

SDR\_HARDFAIL SDR\_BASE + 0x00000004 // 运算模块无响应

SDR\_OPENDEVICE SDR\_BASE + 0x00000005 // 打开设备失败

SDR\_OPENSESSION SDR\_BASE + 0x00000006 // 创建会话失败

SDR\_PARDENY SDR\_BASE + 0x00000007 // 无私钥使用权限

SDR\_KEYNOTEXIST SDR\_BASE + 0x00000008 // 不存在的密钥调用

SDR\_ALGNOTSUPPORT SDR\_BASE + 0x00000009 // 不支持的算法调用

SDR\_ALGMODNOTSUPPORT SDR\_BASE + 0x0000000A // 不支持的算法式调用

SDR\_PKOPERR SDR\_BASE + 0x0000000B // 公钥运算失败

SDR\_SKOPERR SDR\_BASE + 0x0000000C // 私钥运算失败

SDR\_SIGNERR SDR\_BASE + 0x0000000D // 签名运算失败

SDR\_VERIFYERR SDR\_BASE + 0x0000000E // 验证签名失败

SDR\_SYMOPERR SDR\_BASE + 0x0000000F // 对称算法运算失败

SDR\_STEPERR SDR\_BASE + 0x00000010 // 多步运算步骤错误

SDR\_FILESIZEERR SDR\_BASE + 0x00000011 // 文件长度超出限制

SDR\_FILENOEXIST SDR\_BASE + 0x00000012 // 指定的文件不存在

SDR\_FILEOFSERR SDR\_BASE + 0x00000013 // 文件起始位置错误

SDR\_KEYTYPEERR SDR\_BASE + 0x00000014 // 密钥类型错误

SDR\_KEYERR SDR\_BASE + 0x00000015 // 密钥错误

SDR\_ENCDATAERR SDR\_BASE + 0x00000016 // ECC密钥密文

SDR\_RANDERR SDR\_BASE + 0x00000017 // 随机数产生错误

SDR\_PRKRERR SDR\_BASE + 0x00000018 // 私钥使用权限获取失败

SDR\_MACERR SDR\_BASE + 0x00000019 // MAC运算失败

SDR\_FILEEXITS SDR\_BASE + 0x0000001A // 指定的文件已经存在

SDR\_FILEWRITEERR SDR\_BASE + 0x0000001B // 文件写错误

SDR\_NUBUFFER SDR\_BASE + 0x0000001C // 存储空间不足

SDR\_INARGERR SDR\_BASE + 0x0000001D // 输入参数错误

SDR\_OUTARGERR SDR\_BASE + 0x0000001E // 输出参数错误

SDR\_HANDLE\_CLOSED SDR\_BASE + 0x0000001F //函数执行过程中，该会话已经关闭

SDR\_HANDLE\_COUNT SDR\_BASE + 0x00000020 //打开的会话太多

SDR\_HANDLE\_INVALID SDR\_BASE + 0x00000021 //指定的会话句柄无效

SDR\_LOGIN\_FAILED SDR\_BASE + 0x00000022 //获取私钥使用权限失败

SDR\_LOGIN\_REPEAT SDR\_BASE + 0x00000023 //获取私钥使用权限重复

SDR\_NOT\_LOGIN SDR\_BASE + 0x00000024 //私钥使用权限未获取

SDR\_INPUT\_LEN\_ERROR SDR\_BASE + 0x00000025 //输入参数长度指示错误

SDR\_KEYID\_INVALID SDR\_BASE + 0x00000026 //指定的密钥号非法

SDR\_MECHANISM\_INVALID SDR\_BASE + 0x00000027 //机制无效

SDR\_NOT\_INITIALIZED SDR\_BASE + 0x00000028 //未调用初始化

SDR\_ALREADY\_INITIALIZED SDR\_BASE + 0x00000029 //初始化已调用

SDR\_DEVICEHANDLE\_INVALID SDR\_BASE + 0x0000002A //设备句柄无效

SDR\_DEVICE\_ERROR SDR\_BASE + 0x0000002B

SDR\_KEY\_MEM\_FULL SDR\_BASE + 0x0000002C

SDR\_KEY\_GEN\_FAIL SDR\_BASE + 0x0000002D //产生密钥失败

SDR\_FILE\_OPR\_ERR SDR\_BASE + 0x0000002E

SDR\_KEY\_NO\_EXIST SDR\_BASE + 0x0000002F

SDR\_MALLOC\_ERR SDR\_BASE + 0x00000030

SDR\_DATA\_INVALID SDR\_BASE + 0x00000031

SDR\_SM2\_KEYBITS SDR\_BASE + 0x00000032

SDR\_SESSION\_INIT SDR\_BASE + 0x00000033

SDR\_RANDOM\_GEN SDR\_BASE + 0x00000034 //产生随机数

SDR\_KEYPAIR\_LENGTH SDR\_BASE + 0x00000035 //非对称密钥的指数长度

SDR\_PADDING\_RSA SDR\_BASE + 0x00000036 // RSA padding 错误

SDR\_UNPADDING\_RSA SDR\_BASE + 0x00000037 // RSA unpadding 错误

SDR\_HANDLE\_SYMKEY\_INVALID SDR\_BASE + 0x00000038 // 会话密钥句柄无效

SDR\_READ\_INI\_ERR SDR\_BASE + 0x00000039 // 读取配置文件失败

SDR\_HEADBEAT\_TEST\_ERR SDR\_BASE + 0x00000040 // 连接密码机心跳测试失败

SDR\_DATA\_TRANS\_ERR SDR\_BASE + 0x00000041 // 数据传输错误

SDR\_DATA\_DEVINFO\_ERR SDR\_BASE + 0x00000042 // 设备信息获取失败

SDR\_DATA\_GEN\_RANDOM\_ERR SDR\_BASE + 0x00000043 // 随机数生成失败

SDR\_DATA\_INDEX\_OUT\_ERR SDR\_BASE + 0x00000044 // 索引超出范围

SDR\_HANDLE\_INPUT\_INVALID SDR\_BASE + 0x00000045 // 输入的句柄无效

SDR\_DATA\_PRI\_ACCESS\_ERR SDR\_BASE + 0x00000046 // 获取内部私钥权限失败

SDR\_DATA\_EXP\_PUBKEY\_ERR SDR\_BASE + 0x00000047 // 导出公钥失败

SDR\_DATA\_GEN\_SESKEY\_ERR SDR\_BASE + 0x00000048 // 生成会话密钥失败

SDR\_DATA\_DIGI\_ENEV\_ERR SDR\_BASE + 0x00000049 // 数字信封转换失败

SDR\_DATA\_GEN\_AGREE\_KEY\_ERR SDR\_BASE + 0x00000050 // 生成协商密钥失败

SDR\_DATA\_IM\_RSA\_KEY\_ERR SDR\_BASE + 0x00000051 // 导入RSA密钥对失败

SDR\_DATA\_IM\_ECC\_KEY\_ERR SDR\_BASE + 0x00000052 // 导入ECC密钥对失败

SDR\_DATA\_IM\_ECC\_ENC\_ERR SDR\_BASE + 0x00000053 // ECC公钥加密失败

SDR\_DATA\_IV\_SYM\_ERR SDR\_BASE + 0x00000054 // IV数据错误

SDR\_DATA\_HASH\_INIT\_ERR SDR\_BASE + 0x00000055 // 杂凑运算初始化失败

SDR\_DATA\_HASH\_OP\_ERR SDR\_BASE + 0x00000056 // 杂凑运算失败

SDR\_DATA\_CREATE\_FILE\_ERR SDR\_BASE + 0x00000057 // 创建文件失败

SDR\_DATA\_DELETE\_FILE\_ERR SDR\_BASE + 0x00000058 // 删除文件失败

## 5.3 应急广播命令错误码说明

YJERR\_OK 0x00000000 // 操作成功

YJERR\_OPENDEVICE 0x00000100 // 打开会话失败

YJERR\_CLOSEDEVICE 0x00000200 // 关闭会话失败

YJERR\_OPENSESSION 0x00000300 // 打开会话失败

YJERR\_CLOSESESSION 0x00000400 // 关闭会话失败

YJERR\_CREATEFILE 0x00000500 // 创建文件失败

YJERR\_DELETEFILE 0x00000600 // 删除文件失败

YJERR\_READFILE 0x00000700 // 读取文件失败

YJERR\_WRITEFILE 0x00000800 // 写入文件失败

YJERR\_HASHINIT 0x00000900 // HASH初始化失败

YJERR\_HASHUPDATE 0x00000A00 // HASH操作失败

YJERR\_HASHFINAL 0x00000B00 // HASH结束失败

YJERR\_EXPPUBKEY 0x00000C00 // 导出公钥失败

YJERR\_ACCESSPRIKEY 0x00000D00 // 获取私钥权限失败

YJERR\_RELEASESPRIKEY 0x00000E00 // 释放私钥权限失败

YJERR\_SIGNATURE 0x00000F00 // 签名失败

YJERR\_VERIFY 0x00001000 // 验签失败

YJERR\_FILEOPEN 0x0000E100 // 打开本机文件失败

YJERR\_FILEREAD 0x0000E200 // 读取本机文件失败

YJERR\_COUNTERMUTEXFAILED 0x0000E300 // 签名计数器互斥锁操作失败

YJERR\_COUNTER 0x0000E400 // 签名计数器操作失败

YJERR\_INDEXMUTEXFAILED 0x0000E500 // 证书索引文件互斥锁操作失败

YJERR\_INDEX 0x0000E600 // 证书索引文件操作失败

YJERR\_MEMSMALL 0x0000E700 // 提供的缓冲区指针过小

YJERR\_CERTINDEX 0x0000E800 // 证书索引超限

YJERR\_NOCERT 0x0000E900 // 无证书

YJERR\_CERTCOUNTMAX 0x0000EA00 // 证书数量已经最大

YJERR\_STRLEN 0x0000EB00 // 字符串长度必须为偶数

YJERR\_STRVALUE 0x0000EC00 // 字符串值必须为0-9或A-F