薄膜卡安卓接口规范

目录

1 预植证书卡片指令集

1.1 获取设备信息

CLA	INS	P1	P2	P3	Data	备注
B0	10	00	00	00		表示不知道设备信息长度，通过卡片返回 SW=6CXX 中的 XX 得到真实长度
B0	10	00	00	XX	设备信息	XX 为上一条指令后，卡片返回 SW=6CXX 中的 XX 值 设备信息域为 DEVINFO 结构

1.2 校验 PIN

CLA	INS	P1	P2	P3	Data	备注
B0	1D	0Y	0X	Lc	PIN 码/无	P1 中 0Y 表示： 00——校验 PIN Lc 为 PIN 码长度 01——获取剩余重试次数 Lc=0，卡片返回 63CX P2 中 0X 表示 PIN 码角色，例如用户角色 0x01

1.3 创建文件

CLA	INS	P1	P2	P3	Data	备注
B0	E0	00	00	Lc	文件属性结构	详见 SIM_FILE

1.4 读文件

CLA	INS	P1	P2	P3	Data	备注
B0	A4	00	0C	02	文件 id	先选择要读取的文件
B0	B0	XX	XX	Le	读取的文件内容	P1P2 表示读取的起始位置 Le 表示要读取的内容长度

1.5 写文件

CLA	INS	P1	P2	P3	Data	备注
B0	A4	00	0C	02	文件 id	先选择要写入的文件
B0	D6	XX	XX	Le	写入的文件内容	P1P2 表示写入的起始位置 Le 表示要写入的内容长度

1.6 卡内产生密钥对

CLA	INS	P1	P2	P3	Data	备注
B0	26	0X	00	04	公钥文件 id+私钥文件 id	P1 表示密钥类型： 01——RSA1024 密钥对 02——RSA2048 密钥对 03——SM2 密钥对 公钥文件 id+私钥文件 id 表示生成密钥对后写入该文件 注：若要读取公钥信息请继续使用读文件接口

1.7 RSA 公私钥运算

CLA	INS	P1	P2	P3	Data	备注
B0	17	0Y	0X	Lc	公/私钥文件 id+	P1 表示待计算数据的分类：

					待计算数据	00——已做好填充，直接进行 RSA 运算 01——未填充，需对待计算数据进行 PKCS#1 补位（卡片固定 prepend “0001FF……FF00”，填满 1024/2048 bits），这样可使 RSA 私钥签名时减少下发的数据量，提高业务响应速度 P2 表示输入数据分包： 00——唯一包，开头 2 字节固定表示要使用的 RSA 公/私钥文件 id，后续再紧随待计算数据 01——多包的首包，开头 2 字节固定表示要使用的 RSA 公/私钥文件 id，后续再紧随待计算数据 02——多包的中间包 03——多包的尾包 Lc 表示该包的输入数据长度 当卡片正常接收完首包或中间包时，返回 SW=9000；接收完尾包并计算完毕后，返回 SW=61XX （当使用安卓 SDK 接口时，SDK 会在内部自动发送 Get Response 指令并返回响应）
B0	C0	00	00	XX	运算结果	XX 为上一条指令后，卡片返回 SW=61XX 中的 XX 值 通过该指令获取运算结果，直至卡片返回 SW=9000 表示响应完

						毕

1.8 设置/获取 SM2 算法参数

CLA	INS	P1	P2	P3	Data	备注
B0	1A	0Y	0X	Lc	参数值	P1 表示： 00——设置参数 01——获取参数 P2 表示设置的参数类型： 01——身份标识，此时 Lc 为身份标识长度，输入参数为身份标识 02——曲线参数，此时 Lc=sizeof（SIM_SM2_PARAM），输入参数为 SIM_SM2_PARAM 结构

1.9 SM2 公私钥运算

CLA	INS	P1	P2	P3	Data	备注
B0	2A	00	0X	Lc	输入数据	P2 表示输入数据分包： 00——唯一包，开头 2 字节固定表示要使用的 SM2 公/私钥文件 id，后续再紧随待计算数据 01——多包的首包，开头 2 字节固定表示要使用的 SM2 公/私钥文件 id，后续再紧随待计算数据 02——多包的中间包 03——多包的尾包 Lc 表示该包的输入数据长度 当卡片正常接收完首包或中间

						包时，返回 SW=9000；接收完尾包并计算完毕后，返回 SW=61XX （当使用安卓 SDK 接口时，SDK 会在内部自动发送 Get Response 指令并返回响应）
B0	C0	00	00	XX	运算结果	XX 为上一条指令后，卡片返回 SW=61XX 中的 XX 值 通过该指令获取运算结果，直至卡片返回 SW=9000 表示响应完毕

2 应用接口卡片指令集

2.1 选择文件

CLA	INS	P1	P2	P3	Data	备注
B0	A4	00	0C	02	文件 id	

2.2 删除文件

CLA	INS	P1	P2	P3	Data	备注
B0	04	00	00	02	文件 id	

2.3 取文件属性

CLA	INS	P1	P2	P3	Data	备注
B0	B1	XX	XX	Le	文件属性结构	P1P2 表示要查看的文件 id Le=sizeof（SIM_FILE） 文件属性域为 SIM_FILE 结构

2.4 卡内产生随机数

CLA	INS	P1	P2	P3	Data	备注
B0	12	00	00	Le	随机数	Le 为需要获取的随机数长度

2.5 SM2 签名

CLA	INS	P1	P2	P3	Data	备注
B0	2C	00	00	22	私钥文件 id（2bytes）+摘要数据（32bytes）	开头 2 字节固定表示要使用的 SM2 私钥文件 id，后面紧随的是摘要数据 （注：摘要数据 e=SM3(Za||M)，否则卡片无法计算签名） 卡片返回 SW=61XX （当使用安卓 SDK 接口时，SDK 会在内部自动发送 Get Response 指令并返回响应）
B0	C0	00	00	XX	运算结果	XX 为上一条指令后，卡片返回 SW=61XX 中的 XX 值

2.6 SM4 加解密

CLA	INS	P1	P2	P3	Data	备注
B0	24	0X	0Y	Lc	密钥文件 id+ICV（可选）+待加解密数据	P1 中 0X 表示： 00——ECB 模式（不存在 ICV 域） 01——CBC 模式（存在 ICV 域） P2 中 0Y 表示： 00——加密 01——解密

						输入域开头 2 字节固定表示要使用的 SM4 密钥文件 id，后续为 16 字节 ICV（可选），最后为 16 字节整数倍的待加解密数据 卡片返回状态字 61XX （当使用安卓 SDK 接口时，SDK 会在内部自动发送 Get Response 指令并返回响应）
B0	C0	00	00	XX	加解密数据	XX 为上一条指令后，卡片返回 SW=61XX 中的 XX 值 通过该指令获取运算结果，直至卡片返回 SW=9000 表示响应完毕

2.7 摘要运算

CLA	INS	P1	P2	P3	Data	备注
B0	18	0X	0Y	Lc	消息数据	P1 中 0X 表示摘要算法： 01——SHA1 03——SM3 P2 表示输入数据分包： 00——唯一包 01——多包的首包 02——多包的中间包 03——多包的尾包 卡片返回状态字 61XX （当使用安卓 SDK 接口时，SDK 会在内部自动发送 Get Response 指令并返回响应）

B0	C0	00	00	XX	摘要数据	XX 为上一条指令后，卡片返回 SW=61XX 中的 XX 值 通过该指令获取运算结果，直至卡片返回 SW=9000 表示响应完毕

2.8 SM2 签名验证

CLA	INS	P1	P2	P3	Data	备注
B0	21	00	0X	Lc	公钥文件 id+签名值+摘要数据/原始待签名数据	P2 表示输入数据类型： 00——摘要数据，固定 32 字节(e=SM3(Za||M)) 01——原始待签名数据 输入数据开头 2 字节固定表示要使用的 SM2 公钥文件 id，后面紧随的是 64 字节签名值，最后是 32 字节摘要数据或不定长原始待签数据

2.9 解锁 PIN 码

CLA	INS	P1	P2	P3	Data	备注
B0	1F	0Y	XX	Lc	PUK 码 LV+新 PIN 码 LV/新 PUK 码 LV	P1 中 0Y 表示操作类型： 00——解锁 PIN 01——修改解锁码（每个 id 仅能修改一次） P2 表示解锁密钥 id

2.10 修改 PIN 码

CLA	INS	P1	P2	P3	Data	备注
B0	1E	00	0X	Lc	旧 PIN 码 LV+新 PIN 码 LV	P2 中 0X 表示 PIN 码角色，例如用户角色 0x01

2.11 设置应用 HASH 值

CLA	INS	P1	P2	P3	Data	备注
B0	1C	00	00	Lc	sha1 值 1+sha1 值 2（可选）+…… +sha1 值 6(可选)	最大支持填入 6 只 keystore 的 sha1 值,即 Lc 必须为 20 的倍数，且 20<=Lc<=120

3 结构体说明

3.1 文件属性结构

```
#define FILE_ID_LEN         2
typedef struct _SIM_FILE
{
    unsigned char     type;         //文件类型
    unsigned short    room;         //空间大小 文件类型为二进制文件时有效
    unsigned char     read_Acl;     //读取权限 对rsa私钥文件该值无效，卡的私钥不允许读取
    unsigned char     write_Acl;    //写入权限
    unsigned char     use_Acl;      //使用权限当为公私钥文件时有效
    unsigned char     id[FILE_ID_LEN];   //文件ID
} SIM_FILE;
```

3.2 SM2 曲线参数结构

```
#define KEY_LEN_SM2      32
typedef struct _SIM_SM2_PARAM {
    unsigned char p[KEY_LEN_SM2];     //素数p
    unsigned char a[KEY_LEN_SM2];     //系数a
    unsigned char b[KEY_LEN_SM2];     //系数b
    unsigned char n[KEY_LEN_SM2];     //阶
    unsigned char x[KEY_LEN_SM2];     //基点G的x坐标
    unsigned char y[KEY_LEN_SM2];     //基点G的y坐标
} SIM_SM2_PARAM;
```

4 卡片状态字说明

状态字（**Hex**）	含义
9000	成功
6700	P3 错误
6A86	P1P2 错误
6F01	未知错误
6CXX	Le 错误
63CX	认证不通过，剩余 X 次机会
6983	卡片锁死
6982	权限不足
61XX	待获取 XX 字节数据
6989	待写入数据越界
6F00	底层接口运算错误
6F88	创建文件失败（如文件已存在）
698F	未校验 PIN，权限不足
6A82	文件不存在

6A83	算法不支持
6985	MAC 错误
698B	文件类型错误

5 安卓 SDK

5.1 配置与调用

5.1.1 清单配置

清单文件 AndroidManifest.xml 中需申请以下权限：

```
<uses-permission android:name="org.simalliance.openmobileapi.SMARTCARD" />
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_SMS"/>
<uses-permission android:name="android.permission.WRITE_SMS"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.WRITE_CONTACTS"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>

<application
    ……
    <uses-library android:name="org.simalliance.openmobileapi"
android:required="false" />
    ……
</application>
```

5.1.2 工程调用

```
import com. thinsim.model.Card;
import com. thinsim.model.Card.SCSupported;
public class MainActivity extends Activity {
    private Card mCard = new Card();
}
```

5.2 常量定义

错误码	值
XKR _OK	0
XKR_PWD_N	N(N>0)
XKR _IO_FAILED	-2
XKR_BACK_DATA	-4
XKR_KEY_LOCKED	-16

5.3 接口定义

5.3.1 OpenSEService

5.3.1.1 函数功能

每次开启客户端时首先调用此接口开启 SE 通道

5.3.1.2 函数原型

public void OpenSEService(Context context, final SCSupported scSupported)

5.3.1.3 参数说明

context：Context 上下文

5.3.1.4 返回值

scSupported：回调接口

public void isSupported(boolean success)

success——true，支持 OTI 通道

success——false，不支持 OTI 通道

5.3.2 CloseSEService

5.3.2.1 函数功能

每次退出或关闭客户端进程时需调用此接口关闭数据通道。

5.3.2.2 函数原型

public void CloseSEService()

5.3.3 SendAPDU

5.3.3.1 函数功能

使用本接口可依照卡片指令集发送任意 APDU 指令至卡片，实现相应接口。为保证在手机环境下数据传输完整无误，本接口会自动在 APDU 指令尾部填充 MAC 值，故当数据量很大需分包发送时，Lc 长度务必不要超过 247 字节，否则本接口会抛出异常。

5.3.3.2 函数原型

public String SendAPDU(String apdu)

5.3.3.3 参数说明

apdu：APDU 指令

5.3.3.4 返回值

卡片响应

5.3.4 VerifyPIN

5.3.4.1 函数功能

校验用户输入的 PIN 码是否正确。本接口会自动加密敏感数据再传输。

5.3.4.2 函数原型

public int VerifyPIN(int role, byte[] pin)

5.3.4.3 参数说明

role：角色 id

pin：用户输入的 PIN 码

5.3.4.4 返回值

错误码

5.3.5 ChangePIN

5.3.5.1 函数功能

校验用户输入的原 PIN 码，若正确则修改为输入的新 PIN 码。本接口会自动加密敏感数据再传输。

5.3.5.2 函数原型

public int ChangePIN(int role, byte[] oldPIN, byte[] newPIN)

5.3.5.3 参数说明

role：角色 id

oldPIN：用户输入的原 PIN 码

newPIN：用户输入的新 PIN 码

5.3.5.4 返回值

错误码

5.3.6 UnlockPIN

5.3.6.1 函数功能

当 PIN 码错误次数超限，卡片被锁定时，用户可使用 PUK 码进行卡片解锁，并初始化新 PIN 码。本接口会自动加密敏感数据再传输。

5.3.6.2 函数原型

public int UnlockPIN(int pukId, byte[] puk, byte[] pin)

5.3.6.3 参数说明

pukId：PUK 码 id

puk：用户输入的 PUK 码

pin：用户输入的初始化 PIN 码

5.3.6.4 返回值

错误码