**AF-S300 服务器密码机**

**Java语言接口使用规范**

北京神州安付科技股份有限公司

2023-03-16

#### 算法标识和数据结构

**1.1获取设备信息**

|  |  |  |
| --- | --- | --- |
| 设备定义信息 | | |
| 字段名称 | 数据长度（字节） | 含义 |
| IssuerName | 40 | 设备生产厂商名称 |
| DeviceName | 16 | 设备型号 |
| DeviceSerial | 16 | 设备编号  包含：日期（8字符）、批次号（3字符）、流水号（5字符） |
| AsymAlgAbility | 2 |  |
| DeviceVersion |  | 密码设备内部软件的版本号 |
| StandardVersion |  | 密码设备支持的接口规范版本号 |
| SymAlgAbility |  | 所有支持的对称算法，表示方法为对称算法标识按位或运算结果 |
| HashAlgAbility |  | 所有支持的杂凑算法，表示方法为杂凑算法标识按位或运算结果 |
| BufferSize |  | 支持的最大文件储存空间 |

实际定义数据结构定义

public class DeviceInfo implements AFStruct {

private byte[] IssuerName = new byte[40];

private byte[] DeviceName= new byte [16];

private byte[] DeviceSerial = new byte[16];

private int DeviceVersion;

private int StandardVersion;

private int[] AsymAlgAbility = new int[2];

private int SymAlgAbility;

private int HashAlgAbility;

private int BufferSize;

}

###### 1.2 查询证书信任列表别名

|  |  |  |
| --- | --- | --- |
| 查询证书信任列表别名的数据结构定义 | | |
| 字段名称 | 数据长度（字节） | 含义 |
| certList |  |  |
| number |  |  |

实际定义数据结构定义

public class certAltNameTrustList {

private byte[] certList;

private int number;

}

###### 1.3 解码数字信封

1.3.1解码PKCS7格式的带签名的数字信封数据

|  |  |  |
| --- | --- | --- |
| 解码PKCS7格式的带签名的数字信封数据的数据结构定义 | | |
| 字段名称 | 数据长度（字节） | 含义 |
| data |  | 原始数据 |
| signerCertificate |  | 签名者证书 |
| digestAlgorithm |  | HASH算法 |
| signedData |  | Base64编码的签名值 |

实际定义数据结构定义

public class AFPkcs7DecodeData {

private byte[] data;

private byte[] signerCertificate;

private int digestAlgorithm;

private byte[] signedData;

}

1.3.2 解码基于SM2算法的带签名的数字信封数据

|  |  |  |
| --- | --- | --- |
| 解码基于SM2算法的带签名的数字信封数据的数据结构定义 | | |
| 字段名称 | 数据长度（字节） | 含义 |
| data |  | 原始数据 |
| signerCertificate |  | 签名者证书 |
| digestAlgorithm |  | HASH算法 |

实际定义数据结构定义

public class AFSM2DecodeSignedAndEnvelopedData {

private byte[] data;

private byte[] signerCertificate;

private int digestAlgorithm;

}

###### 1.4 解码基于SM2算法的签名数据

|  |  |  |
| --- | --- | --- |
| 解码基于SM2算法的签名数据的数据结构定义 | | |
| 字段名称 | 数据长度（字节） | 含义 |
| signedData |  | Base64编码的签名值 |

实际定义数据结构定义

public class AFSM2DecodeSignedData extends AFSM2DecodeSignedAndEnvelopedData {

private byte[] signedData;

}

#### 服务器密码机接口

* 1. 获取设备

AFHsmDevice device = new AFHsmDevice.Builder("192.168.10.40", 8008, "abcd1234") // 服务器IP、端口、设备密钥为必须参数  
  
 //以下为可选参数 根据实际情况设置  
 .connectTimeOut(10000) // 连接超时时间，默认5秒  
 .responseTimeOut(10000) // 响应超时时间，默认10秒  
 .retryCount(3) // 重试次数，默认3次  
 .retryInterval(1000) // 重试间隔，默认1秒  
 .bufferSize(1024 \* 1024) // 缓冲区大小，默认1M  
 .channelCount(10) // 通道数，默认10个

.build();

#### 设备接口描述

###### 3.1 获取随机数

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] getRandom(int length) throws AFCryptoException; | |
| 作用 | 获取随机数，用户获取指定长度的随机数数据 | |
| 调用 | byte[] random = device.getRandom(5); | device说明：  AFHsmDevice device =new AFHsmDevice.Builder(host, port, passwd)  .responseTimeOut(100000)  .build(); |
| 参数 | length | 待取得随机数的长度 |
| 返回值 | 返回取得的随机数数据,Base64编码格式,需要解码后为length长度 | |

###### **3.2获取设备信息**

|  |  |  |
| --- | --- | --- |
| 方法 | public DeviceInfo getDeviceInfo() throws AFCryptoException; | |
| 作用 | 获取设备信息，用户获取密码机的设备信息,查看设备信息时，可使用DeviceInfo中的toString()方法。 | |
| 例如：System.out.println(device.getDeviceInfo().toString()); | |
| 私有参数 | IssuerName | 设备生产厂商名称 |
| DeviceName | 设备型号 |
| DeviceSerial | 设备编号  包含：日期（8字符）、批次号（3字符）、流水号（5字符） |
| DeviceVersion |  |
| StandardVersion | 密码设备内部软件的版本号 |
| AsymAlgAbility | 密码设备支持的接口规范版本号 |
| SymAlgAbility | 所有支持的对称算法，表示方法为对称算法标识按位或运算结果 |
| HashAlgAbility | 所有支持的杂凑算法，表示方法为杂凑算法标识按位或运算结果 |
| BufferSize | 支持的最大文件储存空间 |
| 返回值 | 返回签名验签服务器设备信息 | |

###### **3.3导出公钥**

3.3.1 导出SM2签名公钥

|  |  |  |
| --- | --- | --- |
| 方法 | public SM2PublicKey getSM2SignPublicKey(int index) throws AFCryptoException; | |
| 作用 | 导出SM2签名公钥，导出密码机内部对应索引和用途的SM2公钥信息 | |
| 参数 | index | 密码设备内部存储的SM2索引号 |
| 返回值 | SM2PublicKey | public class SM2PublicKey implements Key {  private int length; // 256/512 模长  private byte[] x = new byte[64]; //公钥x  private byte[] y = new byte[64]; //公钥y  } |

3.3.2导出SM2加密公钥

|  |  |  |
| --- | --- | --- |
| 方法 | public SM2PublicKey getSM2EncryptPublicKey(int index) throws AFCryptoException; | |
| 作用 | 导出SM2加密公钥，导出密码机内部对应索引和用途的SM2公钥信息 | |
| 参数 | index | 密码设备内部存储的SM2索引号 |
| 返回值 | SM2PublicKey | public class SM2PublicKey implements Key {  private int length; // 256/512 模长  private byte[] x = new byte[64]; //公钥x  private byte[] y = new byte[64]; //公钥y  } |

3.3.3导出RSA签名公钥

|  |  |  |
| --- | --- | --- |
| 方法 | public RSAPubKey getRSASignPublicKey(int index) throws AFCryptoException; | |
| 作用 | 导出RAS签名公钥，导出密码机内部对应索引和用途的RSA签名公钥信息 | |
| 参数 | index | 密码设备内部存储的RAS索引号 |
| 返回值 | RSAPubKey | public class RSAPubKey implements IAFStruct {  private int bits;  private byte[] m = new byte[LiteRSARef\_MAX\_LEN];  private byte[] e = new byte[LiteRSARef\_MAX\_LEN];  } |

3.3.4 导出RSA加密公钥

|  |  |  |
| --- | --- | --- |
| 方法 | public RSAPubKey getRSAEncPublicKey(int index) throws AFCryptoException; | |
| 作用 | 导出RAS加密公钥，导出密码机内部对应索引和用途的RSA加密公钥信息 | |
| 参数 | index | 密码设备内部密钥存储的RSA索引号 |
| 返回值 | RSAPubKey | public class RSAPubKey implements IAFStruct {  private int bits;  private byte[] m = new byte[LiteRSARef\_MAX\_LEN];  private byte[] e = new byte[LiteRSARef\_MAX\_LEN];  } |

###### **3.4生成密钥对**

3.4.1生成SM2密钥对

|  |  |  |
| --- | --- | --- |
| 方法 | public SM2KeyPair generateSM2KeyPair(int keyType) throws AFCryptoException ; | |
| 作用 | 生成指定类型的SM2密钥对 | |
| 参数 | keyType | 密钥类型 0:签名密钥对 1:加密密钥对 2:密钥交换密钥对 3:默认密钥对 |
| 返回值 | SM2KeyPair | public class SM2KeyPair {  private int bits = 256; //模长恒为256  //公钥  private SM2PublicKey pubKey;  //私钥  private SM2PrivateKey priKey;  } |

3.4.2生成RSA密钥对

|  |  |  |
| --- | --- | --- |
| 方法 | public RSAKeyPair generateRSAKeyPair(ModulusLength length) throws AFCryptoException ; | |
| 作用 | 生成指定类型的RSA密钥对 | |
| 参数 | length | 模长 1024 | 2048 枚举定义在com.af.constant.ModulusLength中 |
| 返回值 | SM2KeyPair | public class RSAKeyPair {  private RSAPubKey pubKey;  private RSAPriKey priKey;  } |

## **3.5会话密钥相关**

3.5.1 生成会话密钥 非对称加密

|  |  |  |
| --- | --- | --- |
| 方法 | public SessionKey generateSessionKey(Algorithm algorithm, int keyIndex, int length) throws AFCryptoException ; | |
| 作用 | 使用非对称加密方式生成会话密钥 | |
| 参数 | algorithm | 对称算法标识 : SGD\_RSA\_ENC|SGD\_SM2\_2 |
| keyIndex | 用于加密会话密钥的密钥索引 |
| length | 会话密钥长度 8|16|24|32 |
| 返回值 | SessionKey(返回该对象全部字段) | SessionKey {  //会话密钥id  private int id;  //会话密钥长度 字节数  private int length;  //会话密钥  private byte[] key;  } |

3.5.2 导入会话密钥密文 非对称加密

|  |  |  |
| --- | --- | --- |
| 方法 | public SessionKey importSessionKey(Algorithm algorithm, int keyIndex, byte[] key) throws AFCryptoException ; | |
| 作用 | 使用非对称加密方式导入会话密钥密文 | |
| 参数 | algorithm | 对称算法标识 : SGD\_RSA\_ENC|SGD\_SM2\_2 |
| keyIndex | 用于加密会话密钥的密钥索引 |
| key | 会话密钥密文 |
| 返回值 | SessionKey(返回该对象id字段) | SessionKey {  //会话密钥id  private int id;  //会话密钥长度 字节数  private int length;  //会话密钥  private byte[] key;  } |

3.5.3 数字信封转换

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] convertEnvelope(Algorithm algorithm, int keyIndex, byte[] pubKey, byte[] data) throws AFCryptoException ; | |
| 作用 | 数字信封转换,将内部密钥加密的数据转换为指定公钥加密的数据 | |
| 参数 | algorithm | 对称算法标识 : SGD\_RSA\_ENC|SGD\_SM2\_3 |
| keyIndex | 密钥索引 |
| pubKey | 公钥 |
| data | 加密输入信息 |
| 返回值 | 加密输出信息 | |

3.5.4 生成会话密钥（使用对称密钥）

|  |  |  |
| --- | --- | --- |
| 方法 | public SessionKey generateSessionKeyBySym(Algorithm algorithm, int keyIndex, int length) throws AFCryptoException ; | |
| 作用 | 使用对称加密生成会话密钥 | |
| 参数 | algorithm | 对称算法标识 : SGD\_SM1\_ECB|SGD\_SMS4\_ECB |
| keyIndex | 用于加密会话密钥的密钥索引 |
| length | 会话密钥长度 8|16|24|32 |
| 返回值 | SessionKey(返回该对象全部字段) | SessionKey {  //会话密钥id  private int id;  //会话密钥长度 字节数  private int length;  //会话密钥  private byte[] key;  } |

3.5.5 导入会话密钥密文（使用对称密钥）

|  |  |  |
| --- | --- | --- |
| 方法 | public SessionKey importSessionKeyBySym(Algorithm algorithm, int keyIndex, byte[] key) throws AFCryptoException ; | |
| 作用 | 使用对称加密导入会话密钥密文 | |
| 参数 | algorithm | 对称算法标识 : SGD\_SM1\_ECB|SGD\_SMS4\_ECB |
| keyIndex | 用于加密会话密钥的密钥索引 |
| key | 会话密钥密文 |
| 返回值 | SessionKey(返回该对象id length 字段) | SessionKey {  //会话密钥id  private int id;  //会话密钥长度 字节数  private int length;  //会话密钥  private byte[] key;  } |

3.5.6 释放密钥信息

|  |  |  |
| --- | --- | --- |
| 方法 | public void releaseSessionKey(int id) throws AFCryptoException; | |
| 作用 | 根据会话密钥id释放密钥 | |
| 参数 | id | 4字节密钥信息 ID,SessionKey id字段 |

3.5.7 生成协商数据

|  |  |  |
| --- | --- | --- |
| 方法 | public AgreementData generateAgreementData(int keyIndex, ModulusLength length, AgreementData data) throws AFCryptoException; | |
| 作用 | 生成协商数据 | |
| 参数 | keyIndex | 密钥索引 |
| length | 模长 |
| data | 会话密钥密文 |
| 返回值 | AgreementData(返回tempPublicKey publicKey 字段) | public class AgreementData {  /\*\*  \* 会话id  \*/  private int sessionId;  /\*\*  \* 发起方id  \*/  private byte[] initiatorId;  /\*\*  \* 回复方id  \*/  private byte[] responderId;  /\*\*  \* 临时公钥  \*/  private byte[] tempPublicKey;  /\*\*  \* 公钥  \*/  private byte[] publicKey;  } |

3.5.8 生成协商数据及密钥

|  |  |  |
| --- | --- | --- |
| 方法 | public AgreementData generateAgreementDataAndKey(int keyIndex, ModulusLength length, AgreementData data) throws AFCryptoException; | |
| 作用 | 生成协商数据 | |
| 参数 | keyIndex | 密钥索引 |
| length | 模长 |
| data | 会话密钥密文 |
| 返回值 | AgreementData(返回tempPublicKey publicKey 字段) | public class AgreementData {  /\*\*  \* 会话id  \*/  private int sessionId;  /\*\*  \* 发起方id  \*/  private byte[] initiatorId;  /\*\*  \* 回复方id  \*/  private byte[] responderId;  /\*\*  \* 临时公钥  \*/  private byte[] tempPublicKey;  /\*\*  \* 公钥  \*/  private byte[] publicKey;  } |

3.5.9 生成协商密钥

|  |  |  |
| --- | --- | --- |
| 方法 | public AgreementData generateAgreementKey(AgreementData data) throws AFCryptoException; | |
| 作用 | 生成协商数据 | |
|  | data | AgreementData对象 协商数据入参 必须publicKey、tempPublicKey、responderId、 |
| 返回值 | AgreementData(返回sessionId 字段) | public class AgreementData {  /\*\*  \* 会话id  \*/  private int sessionId;  /\*\*  \* 发起方id  \*/  private byte[] initiatorId;  /\*\*  \* 回复方id  \*/  private byte[] responderId;  /\*\*  \* 临时公钥  \*/  private byte[] tempPublicKey;  /\*\*  \* 公钥  \*/  private byte[] publicKey;  } |

**3.6 RSA计算**

3.6.1 RSA内部密钥加密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaInternalEncrypt(int index, byte[] data) throws AFCryptoException; | |
| 作用 | RSA内部密钥加密运算 | |
| index | RSA内部密钥索引 |
| data | 原始数据 |
| 返回值 | 返回加密结果 | |

3.6.2 RSA内部密钥解密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaInternalDecrypt(int index, byte[] data) throws AFCryptoException; | |
| 作用 | RSA内部密钥解密运算 | |
| index | RSA内部密钥索引 |
| data | 加密数据 |
| 返回值 | 返回解密结果 | |

3.6.3 RSA外部密钥加密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaExternalEncrypt(RSAPubKey publicKey, byte[] data) throws AFCryptoException; | |
| 作用 | RSA外部密钥加密运算 | |
| publicKey | RSA公钥信息 |
| data | 原始数据 |
| 返回值 | 返回加密结果 | |

3.6.4 RSA外部密钥解密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaExternalDecrypt(RSAPriKey prvKey, byte[] data) throws AFCryptoException; | |
| 作用 | RSA外部密钥解密运算 | |
| prvKey | RSA私钥信息 |
| data | 原始数据 |
| 返回值 | 返回解密结果 | |

3.6.5 RSA内部密钥签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaInternalSign(int index, byte[] data) throws AFCryptoException; | |
| 作用 | RSA内部密钥签名 | |
| index | RSA内部密钥索引 |
| data | 原始数据 |
| 返回值 | 返回签名值 | |

3.6.6 RSA内部密钥验证签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public boolean rsaInternalVerify(int index, byte[] signedData, byte[] rawData) throws AFCryptoException; | |
| 作用 | RSA内部密钥验证签名 | |
| index | RSA内部密钥索引 |
| signedData | 签名数据 |
| rawData | 原始数据 |
| 返回值 | true: 验证成功，false：验证失败 | |

3.6.7 RSA外部密钥签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] rsaExternalSign(RSAPriKey privateKey, byte[] data) throws AFCryptoException; | |
| 作用 | RSA外部密钥签名 | |
| privateKey | RSA私钥信息 |
| data | 原始数据 |
| 返回值 | 返回签名值 | |

3.6.8 RSA外部密钥验证签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public boolean rsaExternalVerify(RSAPubKey publicKey, byte[] signedData, byte[] rawData) throws AFCryptoException; | |
| 作用 | RSA外部密钥验证签名 | |
| privateKey | RSA私钥信息 |
| signedData | 签名数据 |
| rawData | 原始数据 |
| 返回值 | true: 验证成功，false：验证失败 | |

**3.7 SM2**

3.7.1 SM2内部密钥 加密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2InternalEncrypt(int index, byte[] plain) throws AFCryptoException; | |
| 作用 | SM2内部密钥加密 | |
| index | 密钥索引 |
| plain | 明文数据 |
| 返回值 | 密文数据 | |

3.7.2 SM2内部密钥 解密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2InternalDecrypt(int index, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM2内部密钥解密 | |
| index | 密钥索引 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.7.3 SM2外部密钥 加密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2ExternalEncrypt(SM2PublicKey pubKey, byte[] plain) throws AFCryptoException; | |
| 作用 | SM2外部密钥加密 | |
| pubKey | 公钥信息 |
| plain | 明文数据 |
| 返回值 | 密文数据 | |

3.7.4 SM2外部密钥 解密运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2ExternalDecrypt(SM2PrivateKey prvKey, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM2外部密钥解密 | |
| prvKey | 私钥信息 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.7.5 SM2内部密钥 签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2InternalSign(int index, byte[] data) throws AFCryptoException; | |
| 作用 | SM2内部密钥签名 | |
| index | 密钥索引 |
| data | 待签名数据 |
| 返回值 | 签名数据 | |

3.7.6 SM2内部密钥 验签运算

|  |  |  |
| --- | --- | --- |
| 方法 | public boolean sm2InternalVerify(int index, byte[] data, byte[] sign) throws AFCryptoException; | |
| 作用 | SM2内部密钥验签 | |
| index | 密钥索引 |
| data | 原始数据 |
| sign | 签名数据 |
| 返回值 | 验签结果 | |

3.7.7 SM2外部密钥 签名运算

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm2ExternalSign(SM2PrivateKey prvKey, byte[] data) throws AFCryptoException; | |
| 作用 | SM2外部密钥签名 | |
| prvKey | 私钥信息 |
| data | 原始数据 |
| 返回值 | 签名数据 | |

3.7.8 SM2外部密钥 验签运算

|  |  |  |
| --- | --- | --- |
| 方法 | public boolean sm2ExternalVerify(SM2PublicKey pubKey, byte[] data, byte[] sign) throws AFCryptoException; | |
| 作用 | SM2外部密钥验签 | |
| pubKey | 公钥信息 |
| data | 原始数据 |
| sign | 签名数据 |
| 返回值 | 验签结果 | |

**3.8 对称加密**

3.8.1 SM4 ECB 内部密钥加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4InternalEncryptECB(int keyIndex, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 ECB 内部密钥加密 | |
| 参数 | keyIndex | 密钥索引 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.2 SM4 ECB 外部密钥加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4ExternalEncryptECB(byte[] key, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 ECB 外部密钥加密 | |
| 参数 | key | 密钥 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.3 SM4 ECB 密钥句柄加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4HandleEncryptECB(int keyHandle, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 ECB 密钥句柄加密 | |
| keyHandle | 密钥句柄 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.4 SM4 CBC 内部密钥加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4InternalEncryptCBC(int keyIndex, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 CBC 内部密钥加密 | |
| keyIndex | 密钥索引 |
| iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.5 SM4 CBC 外部密钥加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4ExternalEncryptCBC(byte[] key, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 CBC 外部密钥加密 | |
| key | 密钥索 |
| iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.6 SM4 CBC 密钥句柄加密

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4HandleEncryptCBC(int keyHandle, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM4 CBC 密钥句柄加密 | |
| keyHandle | 密钥句柄 |
| iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.7 SM1 内部密钥加密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1InternalEncryptECB(int keyIndex, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 ECB 内部密钥加密 | |
| keyIndex | 密钥索引 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.8 SM1 外部密钥加密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1ExternalEncryptECB(byte[] key, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 ECB 外部密钥加密 | |
| key | 密钥 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.9 SM1 密钥句柄加密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1HandleEncryptECB(int keyHandle, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 ECB 密钥句柄加密 | |
| keyHandle | 密钥句柄 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.10 SM1 内部密钥加密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1InternalEncryptCBC(int keyIndex, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 CBC 内部密钥加密 | |
| keyIndex | 密钥索引 |
| Iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.11 SM1 外部密钥加密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1ExternalEncryptCBC(byte[] key, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 CBC 外部密钥加密 | |
| key | 密钥 |
| Iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

3.8.12 SM1 密钥句柄加密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1ExternalEncryptCBC(byte[] key, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 CBC 密钥句柄加密 | |
| key | 密钥 |
| Iv | 初始向量 |
| plain | 原始数据 |
| 返回值 | 加密数据 | |

**3.9 对称解密**

3.9.1 SM4 内部密钥 解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4InternalDecryptECB(int keyIndex, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 ECB 内部密钥解密 | |
| 参数 | keyIndex | 密钥索引 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.2 SM4 外部密钥 解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4ExternalDecryptECB(byte[] key, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 ECB 外部密钥解密 | |
| key | 密钥 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.3 SM4 密钥句柄 解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4HandleDecryptECB(int keyHandle, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 ECB 密钥句柄解密 | |
| keyHandle | 密钥句柄 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.4 SM4 内部密钥 解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4InternalDecryptCBC(int keyIndex, byte[] iv, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 CBC 内部密钥解密 | |
| keyIndex | 密钥索引 |
| Iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.5 SM4 外部密钥 解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4ExternalDecryptCBC(byte[] key, byte[] iv, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 CBC 外部密钥解密 | |
| key | 密钥 |
| Iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.6 SM4 密钥句柄 解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm4HandleDecryptCBC(int keyHandle, byte[] iv, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM4 CBC 内部密钥解密 | |
| 参数 | keyHandle | 密钥句柄 |
| Iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.7 SM1 内部密钥解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1InternalDecryptECB(int keyIndex, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM1 ECB 内部密钥解密 | |
| 参数 | keyIndex | 密钥索引 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.8 SM1 外部密钥解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1ExternalDecryptECB(byte[] key, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM1 ECB 外部密钥解密 | |
| 参数 | key | 密钥 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.9 SM1 密钥句柄 解密 ECB

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1HandleDecryptECB(int keyHandle, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM1 ECB 密钥句柄解密 | |
| 参数 | keyHandle | 密钥句柄 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.10 SM1 内部密钥解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1InternalDecryptCBC(int keyIndex, byte[] iv, byte[] plain) throws AFCryptoException; | |
| 作用 | SM1 CBC内部密钥解密 | |
| 参数 | keyIndex | 密钥索引 |
| iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.11 SM1 外部密钥解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1ExternalDecryptCBC(byte[] key, byte[] iv, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM1 CBC外部密钥解密 | |
| 参数 | key | 密钥 |
| iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

3.9.12 SM1 密钥句柄 解密 CBC

|  |  |  |
| --- | --- | --- |
| 方法 | public byte[] sm1HandleDecryptCBC(int keyHandle, byte[] iv, byte[] cipher) throws AFCryptoException; | |
| 作用 | SM1 CBC密钥句柄解密 | |
| 参数 | keyHandle | 密钥句柄 |
| iv | 初始向量 |
| cipher | 密文数据 |
| 返回值 | 明文数据 | |

**3.10 对称批量加密**

3.10.1 SM4 内部批量加密 ECB

3.10.2 SM4外部批量加密 ECB

3.10.3 SM4 密钥句柄批量加密 ECB

3.10.4 SM4 内部批量加密 CBC

3.10.5 SM4 外部批量加密 CBC

3.10.6 SM4 密钥句柄批量加密 CBC

3.10.7 SM1 内部批量加密 ECB

3.10.8 SM1 外部批量加密 ECB

3.10.9 SM1 密钥句柄批量加密 ECB

3.10.10 SM1 内部密钥批量加密 CBC

3.10.11 SM1 外部密钥批量加密 CBC

3.10.12 SM1 密钥句柄批量加密 CBC

**3.11 对称批量解密**

3.11.1 SM4 内部批量解密 ECB

3.11.2 SM4 外部批量解密 ECB

3.11.3 SM4 密钥句柄批量解密 ECB

3.11.4 SM4 内部批量解密 CBC

3.11.5 SM4 外部密钥批量解密 CBC

3.11.6 SM4 密钥句柄批量解密 CBC

3.11.7 SM1 内部密钥批量解密 ECB

3.11.8 SM1 外部密钥批量解密 ECB

3.11.9 SM1 密钥句柄批量解密 ECB

3.11.10 SM1 内部密钥批量解密 CBC

3.11.11 SM1 外部密钥批量解密 CBC

3.11.12 SM1 密钥句柄批量解密 CBC

**3.12 MAC计算**

3.12.1 SM4 计算MAC 内部密钥

3.12.2 SM4 计算MAC 外部密钥

3.12.3 SM4 计算MAC 密钥句柄

3.12.4 SM1 计算MAC 内部密钥

3.12.5 SM1 计算MAC 外部密钥

3.12.6 SM1 计算MAC 密钥句柄

3.12.7 SM3-HMAC

**3.13 Hash计算**

3.13.1 Hash init

3.13.2 Hash init 带公钥

3.13.3 Hash update

3.13.4 Hash doFinal

3.13.5 SM3 Hash

3.13.6 SM3 Hash 带公钥

**3.14 文件操作**

3.14.1 创建文件

3.14.2 读取文件

3.14.3 写入文件

3.14.4 删除文件

**3.15 获取内部对称密钥句柄**