云核客户端安全组件
（鸿蒙版）
用户手册

北京云核网络技术有限公司

2025 年 1 月 30 日

文档修改记录

版本	内容	编写人	编写日期	审核人	审核日期
1.0.0	初版	changbf	2024-06-20		
1.0.1	条件私服打包	changbf	2024-08-30		
1.0.2	解决 ts 数据传输崩溃问题	changbf	2024-10-20		
1.0.3	修复 napi_get_arraybuffer_info 崩溃	changbf	2024-11-02		
1.1.0	正式版本发布	changbf	2024-12-25		
1.1.1	增加 RSA 加解密、签名验签	changbf	2024-12-28		
1.1.2	增加-fstack-protector 参数编译	changbf	2025-01-05		

目　录

1. 引言

1.1 编写目的

“云核客户端安全组件”（PacketCrypto）提供了使用国密算法对交易报文进行加密/解密的能力，具有支持多平台的特性，本次设计以 SDK 的模式在鸿蒙系统上使用。

1.2 背景知识及参考资料

假定开发人员对下列技术有一定的理解：

技术	有关内容
鸿蒙 SDK	鸿蒙 ETS 的基本知识
NAPI-node	实现 ArkTS/TS/JS 和 C/C++之间的交互

1.3 使用环境

鸿蒙星河版。

2. 概述

2.1 系统构成

云核客户端安全组件鸿蒙版本以 HAR 格式提供，由两部分构成：

安全键盘核心实现模块：由 C++实现的 SO 动态库。

安全输入框模块：由 TS 实现。

2.2 功能特点

“云核客户端安全组件”提供了 SM2 加解密、SM4 加解密、HMAC（SM3、SHA1、SHA2、MD5）计算、SM4-SM2 数字信封等功能。支持：

- 鸿蒙 Next API 调用
- PacketCrypto 帮助类调用。用于客户端与服务端报文交互的场景下简化条用，即：SM4、HMAC 随机 Key 由帮助类内部随机生成。
- 国密 SM2 加密/解密
- 国密 SM3 摘要计算
- 国密 SM4 ECB/CBC
- HMAC 计算

- 国际 RSA 加密/解密
- 国际 AES ECB/CBC 加密/解密
- 生物识别算法

2.3 接口描述

2.3.1 生物识别

Biometric 类

2.3.1.1 setCustomBankId

功能说明： 银行加密算法定制，不设置为通用算法
函数原型：
setCustomBankId(customBankId: string): void

参数说明：
[in] customBankId 银行缩写，比如 TCRCB
返回值：
返回值 无

2.3.1.2 setDigestHash

功能说明： 设置摘要算法，支持 SM3 和 SHA1
函数原型：
setDigestHash(digest: string): void

参数说明：
[in] digest 算法 SM3 / SHA1，默认 SM3
返回值：
返回值 无

2.3.1.3 setAsyncAlgorithm

功能说明： 设置非对称算法，支持 SM2 和 RSA
函数原型：
setAsyncAlgorithm(algorithm: string): void

参数说明：
[in] algorithm 支持算法 SM2 / RSA，默认 SM2
返回值：
返回值 无

2.3.1.4 getGestureState

功能说明： 判断本机是否设置了手势密码 G GESTURE P FINGERPRINT F FACE

函数原型：

getGestureState(userId: string): boolean

参数说明：

[in] userId 用户唯一标识(绑定设备模式调用)

返回值：

返回值 true / false

2.3.1.5 removeGestureSecretKey

功能说明： 清除本机手势密码内置的加密因子

函数原型：

removeGestureSecretKey(userId: string): void

参数说明：

[in] userId 用户唯一标识(绑定设备模式调用)

返回值：

返回值 无

2.3.1.6 getGestureWorkKeyWithType

功能说明： 获取手势密码的工作密钥

函数原型：

getGestureWorkKeyWithType(factor: string, gesturePass: string, appPublicKey: string, encPubLicKey: string, timestamp: string, userId: string): string

参数说明：

[in] factor 加密因子

[in] gesturePass 手势源数据

[in] appPublicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥

[in] encPubLicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥

[in] timestamp 时间戳

[in] userId 用户唯一标识(绑定设备模式调用)

返回值：

返回值 工作密钥密文

2.3.1.7 getGestureAuthCodeWithType

功能说明： 获取手势密码的授权码

函数原型：

getGestureAuthCodeWithType(factor: string, gesturePass: string, challengeCode: string, appPublicKey: string, encPublicKey: string, timestamp: string, userId: string): string

参数说明：

[in] factor 加密因子

[in] gesturePass 手势源数据

[in] challengeCode 挑战码(base64)

[in] appPublicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥

[in] encPubLicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥
[in] timestamp 时间戳
[in] userId 用户唯一标识(绑定设备模式调用)
返回值：
返回值 授权码密文

2.3.1.8 getBiometricState

功能说明： 判断本机是否设置指纹密码
函数原型：
getBiometricState(userId: string, type: EnumBiometricType): boolean 参数说明：
参数说明：
[in] userId 用户唯一标识(绑定设备模式调用)
[in] type 人脸 / 指纹
返回值：
返回值 true / false

2.3.1.9 removeBiometricSecretKey

功能说明： 清除指纹密码内置加密因子
函数原型：
removeBiometricSecretKey(userId: string, type: EnumBiometricType): void
参数说明：
[in] userId 用户唯一标识(绑定设备模式调用)
[in] type 人脸 / 指纹
返回值：
返回值 无

2.3.1.10 getBiometricWorkKeyWithPublicKey

功能说明： 获取指纹密码的工作密钥
函数原型：
getBiometricWorkKeyWithPublicKey(appPublicKey: string, encPublicKey: string,
timestamp: string, userId: string, type: EnumBiometricType): string
参数说明：
[in] appPublicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥
[in] encPubLicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥
[in] userId 用户唯一标识(绑定设备模式调用)
[in] type 人脸 / 指纹
返回值：
返回值 工作密钥密文

2.3.1.11 getBiometricAuthCodeWithChallengeCode

功能说明： 获取指纹密码的工作密钥

函数原型：

getBiometricWorkKeyWithPublicKey(appPublicKey: string, encPublicKey: string, timestamp: string, userId: string, type: EnumBiometricType): string

参数说明：

[in] appPublicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥

[in] encPubLicKey 公钥 根据 setAsyncAlgorithm 这是模数或者 ECC 公钥

[in] userId 用户唯一标识(绑定设备模式调用)

[in] type 人脸 / 指纹

返回值：

返回值 工作密钥密文

2.3.1.12 lastErrorMessage

功能说明： 获取错误信息

函数原型：

lastErrorMessage(): string

参数说明：

无

返回值：

返回值 最后一次错误信息

2.3.2 AES 算法

CryptoAES 类

2.3.2.1 encryptAESECB

功能说明： 使用 ECB 模式进行 AES 加密

函数原型：

encryptAESECB(key: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] key 秘钥

[in] message 待加密数据

返回值：

返回值 密文

2.3.2.2 decryptAESECB

功能说明： 使用 ECB 模式进行 AES 解密

函数原型：

decryptAESECB (key: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] key 秘钥

[in] message 密文

返回值：

返回值	解密明文

2.3.2.3 encryptAESCBC

功能说明： 使用 CBC 模式进行 AES 加密

函数原型：

encryptAESCBC(key: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] key	秘钥
[in] message	待加密数据

返回值：

返回值	密文

2.3.2.4 decryptAESCBC

功能说明： 使用 CBC 模式进行 AES 解密

函数原型：

decryptAESCBC (key: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] key	秘钥
[in] message	密文

返回值：

返回值	解密明文

2.3.3 业务加密

CryptoBiz 类

2.3.3.1 getCiphertext

功能说明： 根据算法 algorithm 类型，进行加密

函数原型：

getCiphertext (algorithm: string, pubKey: string, timestamp: string, plaintext: string): string

参数说明：

[in] algorithm	SM2 / RSA
[in] pubKey	SM2 公钥 或者 RSA 模数 16 进制字符串
[in] timestamp	时间戳
[in] plaintext	明文字符串

返回值：

返回值	密文数据，通常是 base64

2.3.4 RSA 算法

CryptoRSA 类

2.3.4.1 encryptRSAWithPKCS1

功能说明： RSA 加密，使用 PKCS1 填充

函数原型：

encryptRSAWithPKCS1(modulus: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] modulus RSA 模数

[in] message 待加密数据

返回值：

返回值 密文数据

2.3.4.2 decryptRSAWithPKCS1

功能说明： RSA 加密，使用 PKCS1 填充

函数原型：

decryptRSAWithPKCS1(modulus: Uint8Array, exponent: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] modulus RSA 私钥模数

[in] exponent RSA 私钥指数

[in] message 加密数据

返回值：

返回值 解密数据

2.3.4.3 encryptRSAWithOAEP

功能说明： RSA 加密，使用 OAEP 填充

函数原型：

encryptRSAWithOAEP(modulus: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] modulus RSA 模数

[in] message 待加密数据

返回值：

返回值 密文数据

2.3.4.4 decryptRSAWithOAEP

功能说明： RSA 加密，使用 OAEP 填充

函数原型：

decryptRSAWithOAEP(modulus: Uint8Array, exponent: Uint8Array, message: Uint8Array): Uint8Array

参数说明：

[in] modulus RSA 私钥模数

[in] exponent RSA 私钥指数

[in] message 加密数据

返回值：

返回值	解密数据

2.3.4.5 signRSAWithSHA1

功能说明： RSA 签名，使用 SHA1 摘要

函数原型：

```
signRSAWithSHA1(modulus: Uint8Array, exponent: Uint8Array,
    message: Uint8Array): Uint8Array
```

参数说明：

[in] modulus	RSA 模数
[in] exponent	RSA 私钥指数
[in] message	待签名数据

返回值：

返回值	密文数据

2.3.4.6 verifyRSAWithSHA1

功能说明： RSA 验签，使用 SHA1 摘要

函数原型：

```
verifyRSAWithSHA1(modulus: Uint8Array, message: Uint8Array,
    signature: Uint8Array): boolean
```

参数说明：

[in] modulus	RSA 私钥模数
[in] message	加密数据
[in] signature	签名数据

返回值：

返回值	验签成功 / 失败

2.3.5 国密加解密算法

CryptoSM 类

2.3.5.1 getRandom

功能说明： 生成随机数，作为秘钥

函数原型：

```
getRandom(length: number): Uint8Array
```

参数说明：

[in] length	长度

返回值：

返回值	随机数

2.3.5.2 encryptSM4ECB

功能说明： SM4 加密，使用 ECB 模式

函数原型：

```
encryptSM4ECB(sm4Key: Uint8Array, plaintext: Uint8Array): Uint8Array Uint8Array
```

参数说明：

[in] sm4Key sm4 秘钥

[in] plaintext 待加密数据

返回值：

返回值 加密密文 / 空

2.3.5.3 decryptSM4ECB

功能说明： SM4 解密，使用 ECB 模式

函数原型：

decryptSM4ECB(sm4Key: Uint8Array, ciphertext: Uint8Array): Uint8Array

参数说明：

[in] sm4Key sm4 秘钥

[in] ciphertext 密文数据

返回值：

返回值 明文数据 / 空

2.3.5.4 encryptSM4CBC

功能说明： SM4 加密，使用 CBC 模式

函数原型：

encryptSM4CBC(sm4Key: Uint8Array, plaintext: Uint8Array): Uint8Array Uint8Array

参数说明：

[in] sm4Key sm4 秘钥

[in] plaintext 待加密数据

返回值：

返回值 加密密文 / 空

2.3.5.5 decryptSM4CBC

功能说明： SM4 解密，使用 CBC 模式

函数原型：

decryptSM4CBC(sm4Key: Uint8Array, ciphertext: Uint8Array): Uint8Array

参数说明：

[in] sm4Key sm4 秘钥

[in] ciphertext 密文数据

返回值：

返回值 明文数据 / 空

2.3.5.6 encryptSM2

功能说明： SM2 加密

函数原型：

encryptSM2(sm2PublicKey: Uint8Array, plaintext: Uint8Array): Uint8Array

参数说明：

[in] sm2PublicKey SM2 公钥

[in] plaintext 待加密数据

返回值：

返回值 密文数据

2.3.5.7 decryptSM2

功能说明： SM2 解密

函数原型：

decryptSM2(sm2PrivateKey: Uint8Array, ciphertext: Uint8Array): Uint8Array

参数说明：

[in] sm2PrivateKey SM2 私钥 Z

[in] ciphertext 密文数据

返回值：

返回值 密文数据

2.3.6 数据存储

DataStorage 类

2.3.6.1 saveKeyInfo

功能说明： 以 key: value 的形式存储敏感信息

函数原型：

saveKeyInfo(key: string, value: string): number

参数说明：

[in] key 可见字符串， 随机数请使用 hex 或者 base64 格式

[in] value value 可见字符串， 随机数请使用 hex 或者 base64 格式

返回值：

返回值 0 成功，错误码

2.3.6.2 readKeyInfo

功能说明： 以 key: value 的形式读取存储的敏感信息

函数原型：

readKeyInfo(key: string): string

参数说明：

[in] key 可见字符串， 随机数请使用 hex 或者 base64 格式

返回值：

返回值 key 对应的数据

2.3.6.3 removeKeyInfo

功能说明： 以 key: value 的形式删除存储敏感信息

函数原型：

removeKeyInfo(key: string): void

参数说明：

[in] key 可见字符串， 随机数请使用 hex 或者 base64 格式

返回值：

返回值 无

2.3.6.4 saveContent

功能说明： 文件名，默认路径为 context.filedir 目录

函数原型：

saveContent(file: string, content: string): number

参数说明：

[in] file 文件名

[in] content 待存储数据

返回值：

返回值 0 成功，错误码

2.3.6.5 readContent

功能说明： 读取文件中的内容

函数原型：

readContent(file: string): string

参数说明：

[in] file 文件名

返回值：

返回值 数据内容

2.3.7 上下文加密

DataStream 类

2.3.7.1 encryptUpStream

功能说明： 对上行数据报文加密，使用 sm2+sm4 算法，使用 fPaaS 数据格式封装

函数原型：

encryptUpStream: (dataStream: string) => string

参数说明：

[in] dataStream 交易中的 body 数据

返回值：

返回值 base64 编码格式的密文数据

2.3.7.2 decryptDownStream

功能说明： 对下行数据报文解密，使用 sm4 算法，使用 fPaaS 数据格式封装

函数原型：

decryptDownStream: (dataStream: string) => string

参数说明：

[in] dataStream base64 编码格式

返回值：

返回值 下行报文原始数据

2.3.8 摘要算法

Digest 类

2.3.8.1 toSM3

功能说明： SM3 摘要算法

函数原型：

toSM3(message: Uint8Array): Uint8Array

参数说明：

[in] message 原数据

返回值：

返回值 摘要数据

2.3.8.2 toMD5

功能说明： MD5 摘要算法

函数原型：

toMD5(message: Uint8Array): Uint8Array

参数说明：

[in] message 原数据

返回值：

返回值 摘要数据

2.3.8.3 toSHA1

功能说明： SHA1 摘要算法

函数原型：

toSHA1(message: Uint8Array): Uint8Array

参数说明：

[in] message 原数据

返回值：

返回值 摘要数据

2.3.8.4 toSHA256

功能说明： SHA256 摘要算法

函数原型：

toSHA256(message: Uint8Array): Uint8Array

参数说明：

[in] message 原数据

返回值：

返回值 摘要数据

2.3.9 HMAC 算法

HMAC 类

2.3.9.1 getHMAC

功能说明： 使用 rawfile 目录下 ckc(含算法、key 的安全文件)计算 HMAC

函数原型：

getHMAC(message: Uint8Array): string

参数说明：

[in] message 原数据

返回值：

返回值 16 进制字符串

2.3.9.2 getHMACWithKey

功能说明： 根据传入的算法和秘钥，计算 HMAC

函数原型：

getHMACWithKey(algorithm: string, key: string | Uint8Array,
message: Uint8Array): Uint8Array

参数说明：

[in] algorithm 原数据

[in] key 秘钥

[in] message 原数据

返回值：

返回值 HMAC 数据

2.3.10 PacketCrypto

PacketCrypto 类，内部自动生成 SM4、HMAC key， 可获取 KEY 的密文，可通过 getCiphertext 获取上行加密(hmac)数据和通过 getPlaintext 解密下行报文数据。

2.3.10.1 setSM2PublicKey

功能说明： 设置 SM2 公钥，16 进制长度为 128 字节

函数原型：

setSM2PublicKey(publicKey: string): boolean

参数说明：

[in] publicKey 公钥

返回值：

返回值 true / false

2.3.10.2 encryptSM4KeyWithSM2

功能说明： 对 SM4 KEY 使用 SM2 加密，秘钥有类构造中自动生成

函数原型：

encryptSM4KeyWithSM2(): Uint8Array

参数说明：

[in] 无

返回值：

返回值 密文

2.3.10.3 encryptHMACKeyWithSM2

功能说明： 对 HMAC KEY 使用 SM2 加密，秘钥有类构造中自动生成

函数原型：

encryptHMACKeyWithSM2(): Uint8Array

参数说明：

[in] 无

返回值：

返回值 密文

2.3.10.4 encryptSM4

功能说明： 对明文使用 SM4 加密

函数原型：

encryptSM4(plaintext: Uint8Array): Uint8Array

参数说明：

[in] 无

返回值：

返回值 密文

2.3.10.5 decryptSM4

功能说明： 对密文使用 SM4 解密

函数原型：

decryptSM4(ciphertext: Uint8Array): Uint8Array

参数说明：

[in] ciphertext 下行报文密文

返回值：

返回值 明文

2.3.10.6 verifyHMACWihSM3

功能说明： 根据明文、hmac 验证，hmac 使用的是 SM3 摘要算法

函数原型：

verifyHMACWihSM3(plaintext: Uint8Array, hmac: Uint8Array): boolean

参数说明：

[in] plaintext 下行报文

[in] hmac 下行报文 hmac，服务端根据上行报文解密出来的 HMAC key 计算

返回值：

返回值 true / false

2.3.10.7 getHMACWithSM3

功能说明： 使用 SM3 摘要算法计算 HMAC

函数原型：

getHMACWithSM3(plaintext : Uint8Array): Uint8Array

参数说明：

[in] plaintext 上行报文

返回值：

返回值 HMAC 值

2.3.10.8 getCiphertext

功能说明： 获取 json 格式密文字符串

函数原型：

getCiphertext(type: number, plaintext: Uint8Array): string

参数说明：

[in] type 类型

TYPE_SM4 : number = 1 使用 SM4 算法

TYPE_HMAC : number = 2 使用 HMAC 算法

TYPE_SM4_HMAC: number = 3 同时使用 SM4 和 HMAC 算法

[in] plaintext 上行报文

返回值：

返回值

TYPE_SM4: 返回 报文密文和 SM4 秘钥密文 组成的 json 字符串

TYPE_HMAC 返回 报文 HMAC 和 HMAC 秘钥密文 组成的 json 字符串

TYPE_SM4_HMAC 返回 报文密文、报文 HMAC、SM4 秘钥密文、HMAC 秘钥密文 组成的 json 字符串

```
{
    'SM4KeyCipher' : 'SM4 秘钥密文'
    'SM4Cipher' : '报文 SM4 密文'
    'HMACKeyCipher' : 'HMAC 秘钥密文'
```

```
        'HMAC' : '报文 HMAC(SM3)'
    }
```

2.3.10.9 getPlaintext

功能说明： 解密服务端下行报文数据，不支持不含密文数据的 HMAC 类型

函数原型：

```
getPlaintext(type:number, sm4Ciphertext: Uint8Array, hmacHash: Uint8Array):Uint8Array
```

参数说明：

[in] type	类型
[in] sm4Ciphertext	SM4 密文
[in] hmacHash	hmac 值

返回值：

返回值	下行报文

2.3.11 签名算法

Signature 类，使用 rawfile 目录下 ckc(含算法、key 的安全文件)

2.3.11.1 getAlgorithm

功能说明： 使用 rawfile 目录下 ckc(含算法、key 的安全文件)计算 HMAC

函数原型：

```
getAlgorithm(): string
```

参数说明：

[in] 无

返回值：

返回值	当前使用的算法

2.3.11.2 verify

功能说明： 使用 CKC 文件中的算法和 key 对数据签名

函数原型：

```
verify(message: Uint8Array, signedData: string): boolean
```

参数说明：

[in] message	原数据
[in] signedData	签名数据

返回值：

返回值	成功 / 失败

2.3.11.3 verifyData

功能说明： 据算法、key 对数据做验签

函数原型：

```
verifyData(algorithm: string, pubkey: string, message: Uint8Array, signdata: string): boolea
```

n

参数说明：

[in] algorithm	算法，支持 RSA-MD5 / RSA-SHA1 / RSA-SHA256 / SM2-SM3
[in] message	原数据
[in] signData	签名数据

返回值：

返回值	成功 / 失败