电 信 终 端 产 业 协 会 标 准

TAF-WG4-AS0010-V1.0.0:2017

TEE Internal API 测试方法(Java 版)

TEE Internal API Test Method for Java

2017-05-18 发布　　2017-05-30 实施

电信终端产业协会　　发 布

目 次

前言 II
引言 III
1 范围 1
2 规范性引用文件 1
3 缩略语 1
4 目标 2
5 测试环境 2
6 测试方法 2
6.1 java.lang 包 2
6.2 tee.framework 包 2
6.3 tee.trustedstorage 包 7
6.4 tee.cryptography 包 11
6.5 tee.time 包 16
6.6 tee.util 包 17
附 录 A （资料性附录） 文档编写记录 18

前　　言

本标准按照 GB/T 1.1-2009给出的规则起草。

本标准由中国移动通信有限公司研究院提出并归口。

本标准起草单位：中国移动通信有限公司研究院、中国信息通信研究院(工业和信息化部电信研究院)。

本标准主要起草人：任晓明、李征、傅镜艺、刘辉、刘扬。

引　　言

随着移动互联时代的深入，手机处理了越来越多的敏感数据，如证券业务、银行业务、支付业务等，其面临的安全问题也日益突出。TEE安全终端应运而生。可信执行环境（TEE）是与传统的终端应用运行环境相独立的一个可信环境，该环境以终端硬件提供的安全能力为依托，基于微内核OS向应用提供各种安全服务，比如安全显示、安全输入、安全存储、安全启动、数据加密等。对用户来说原有的操作体验不变。

为了保障TA调用TEE安全服务的正确性、可靠性和稳定性，需要检测TEE的技术实现是否遵照《TEE Internal API技术要求(Java版)》规范。

Java Internal API 测试方法(Java 版)

1 范围

本规范测试范围针对《TEE Internal API技术要求（Java版）》中的内容进行测试，测试对象为TEE OS中的InternalAPI接口，如图1阴影部分所示。具体测试内容包括java.lang包、tee.framework包、tee.trustedstorage包、tee.cryptography包、tee.util包中的各个类及其成员方法，本规范针对类中的各个接口提供测试方法，测试各个类的功能是否按照对应的技术规范进行实现。测试方法是建立CA到TA的连接，并由CA向TA发送测试指令， TA根据测试指令调用相关方法执行测试,并将结果返回给CA，CA根据执行结果判断接口是否按照规定实现。

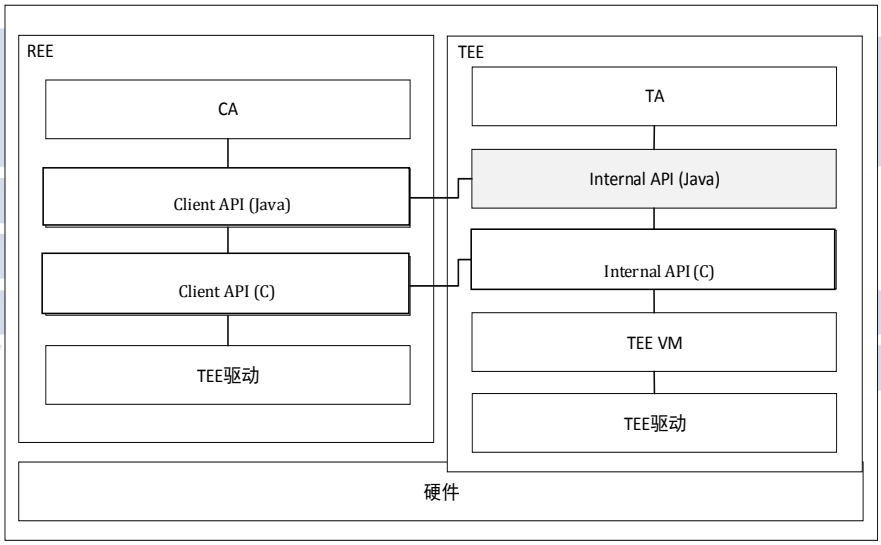

图1 TEE接口架构

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

TAF-WG4-AS0006-V1.0.0:2016　　TEE Internal API技术要求（Java版）

3 缩略语

TA	Trusted Applications	运行在TEE OS并对客户端提供相应服务的可信应用程序

UUID	Universally Unique Identifier	可信应用程序的唯一标识
CA	Client Applications	运行在REE的客户端程序

4 目标

本文件旨在定义TEE Internal API（Java版）的测试方法。

5 测试环境

本规范旨在规范各个类的测试方法，需要根据测试方法开发相应功能的 TA 和 CA。其中 CA 位于 REE 侧，根据测试方法的各步骤需要向 TA 发送不同的测试指令；TA 为安装在 TEE 中的拥有不同 UUID 的可信应用，其根据接收的测试指令调用接口，达到测试相应的接口功能的目的，其具体需要实现的功能在下面的测试方案中将详细说明。

本规范不约束测试工具的实现形式，其可以位于 PC 端或者移动终端，如果测试工具的实现在 PC 端，测试时可通过蓝牙/wifi、usb 线等方式将测试手机连接到 PC，启动测试工具执行测试；如果测试工具的实现在移动终端，其可为 APK 形式，将 APK 安装进手机，启动 APK 执行测试。

6 测试方法

6.1 java.lang 包

6.1.1 Object 类测试

测试编号	6.1.1
测试项目	基础对象类测试
测试目的	验证 object 对象的建立及比较是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开与TA的会话。 3. CA发送指令到TA触发TA执行步骤5-8。 4. TA创建两个不同的Object类对象obj1和obj2。 5. TA比较obj1和obj1。 6. TA比较obj1和obj2。 7. TA将步骤6、7的结果返回给CA。 8. CA调用关闭会话。 9. CA关闭上下文。
预期结果	1. 步骤1-4返回值符合规范要求，表示成功。 2. 步骤8中CA接收的结果和预期结果一致，测试成功。
备注	无

6.2 tee.framework 包

6.2.1 TrustedApplication 类测试

测试编号	6.2.1
测试项目	TA 和 CA 通信建立测试
测试目的	验证 TA 与 CA 通信的建立是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开与TA的会话。 3. CA发送指令到TA触发TA执行步骤4。 4. TA获取CA的指令，并向CA返回任意返回值。 5. CA调用关闭会话。 6. CA关闭上下文。
预期结果	1. 步骤1-4返回值符合规范要求，表示成功。 2. 步骤4中CA能够接收到返回值，测试成功。
备注	无

6.2.2 Parameters 类测试

测试编号	6.2.2
测试项目	通信参数测试
测试目的	验证 CA 与 TA 通信时参数的设置、传递是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，设置参数值，并为其中一个参数申请更大的缓冲区（该指令触发TA执行步骤4、5）。 4. TA获取CA的参数类型、参数值、缓冲区大小。 5. TA基于步骤4获取的值执行指定规则变换为新值并返回给CA。 6. CA获取TA返回的参数值与预期结果比较。 7. CA关闭会话。 8. CA关闭上下文。
预期结果	1. 步骤1-3返回值符合规范要求，表示成功。 2. 步骤6中CA获得的参数值和预期结果一致，测试成功。
备注	无

6.2.3 UUID 类测试

测试编号	6.2.3
测试项目	UUID 类测试
测试目的	验证获取及比较 UUID 的功能是否正确。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开与TA的会话。 3. CA发送指令到TA，触发TA执行步骤4-。

	4. TA获取自身uuid值。 5. TA任意构造的两个不同的UUID类对象uuid1和uuid2。 6. TA比较uuid1和uuid2的uuid值。 7. TA构造UUID类对象uuid3。 8. TA获取uuid3的uuid值。 9. TA返回4、6、8的结果给CA。 10. CA调用关闭会话。 11. CA关闭上下文。
预期结果	1. 步骤1-3返回值为0x00000000，表示成功。 2. 步骤4中CA获取的uuid值为该TA的uuid，测试成功。 3. 步骤6中CA获取到TA的比较结果为不同，测试成功。 4. 步骤8中CA获取的uuid值符合规范对uuid值的编码要求，测试成功。
备注	无

6.2.4 TEESystem 类测试

测试编号	6.2.4
测试项目	TEESystem 类测试
测试目的	验证任务取消标签的设置及任务取消功能是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开与TA1的会话。 步骤3-6验证任务不可取消标识设置及不可取消功能。 3. CA发送指令到TA1触发TA1执行以下任务（该任务会在固定时间t1后给CA返回结果）。 4. TA1获得当前任务取消标识。 5. TA1为当前任务设置不可取消标识，并保持当前任务持续运行。 6. CA在t1时间内发起取消该任务请求。 步骤7- 10验证任务可取消标识及可取消功能。 7. CA发送指令到TA1。 8. TA1创建一个不可取消任务（该任务会在固定时间t1后给CA返回结果）。 9. TA1为该任务设置可以取消标识。 10. CA在时间 t1内发起取消任务请求。 步骤11-13测试panic方法 11. CA保持与TA1的会话（session1），同时打开与TA1的另一个会话（session2）。 12. CA在session1中向TA1发送指令触发步骤13并获取返回值。 13. TA1在时间t2内发起panic请求。 14. CA关闭上下文。
预期结果	1. 步骤1-2返回值符合规范要求，表示成功。 2. 步骤4返回值为True，表示消息标识未屏蔽，与默认情况相同，测试成功。

	3. 步骤6中CA能够获得TA的返回结果，说明任务取消失败，测试成功。 4. 步骤10中CA获得TA返回的错误码，该错误码符合规范要求，测试成功。 5. 步骤11返回值符合规范要求，表示成功。 6. 步骤12中CA获取TA返回的错误码，该错误码符合规范要求，测试成功。
备注	无

6.2.5 InternalClient 接口测试

测试编号	6.2.5
测试项目	TA 与 TA 间通信测试
测试目的	验证 TA 与 TA 通信时参数的设置、传递及异常捕获功能是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文 2. CA打开与TA1的会话。 步骤3-8验证TA间参数设置和传递功能。 3. CA给TA1发送指令触发TA1执行步骤4、5、7、8、9。 4. TA1打开与TA2的会话并约定参数类型。 5. TA1发送指令到TA2触发TA2执行步骤6，设置参数值（遍历值类型和缓冲区类型）。 6. TA2获取TA1的参数值，基于原值执行指定规则变换为新值并返回给TA1。 7. TA1获取到TA2返回的参数值。 8. TA1关闭与TA2的会话。 步骤9-10验证抛出异常功能 9. TA1主动抛出任意异常。 10. CA捕获异常来源并关闭与TA1的会话。 11. CA关闭上下文。
预期结果	1. 步骤1-4返回值符合规范要求，表示成功。 2. 步骤7中TA1获取的参数值与约定的变换规则一致，测试成功。 3. 步骤10中CA获得的错误码符合规范要求，标识异常来源为TA，测试通过。
备注	无

6.2.6 Property 接口测试

测试编号	6.2.6
测试项目	属性返回测试
测试目的	验证获取属性名以及以不同形式返回属性值的功能是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-8。 4. TA接收到CA传入的属性名（遍历所有属性名）。 5. TA获取步骤4中属性名对应的property类对象。

	6. TA使用该property对象获取属性名。- 7. TA按照property对象的类型，获取该property类对象返回的属性值。 8. TA将步骤6、7的结果返回给CA。 9. CA关闭与TA的会话。 10. CA关闭上下文。
预期结果	1. 步骤1-3返回值符合规范要求，表示成功。 2. 步骤6中CA获得的属性名和传入的属性名一致，测试成功。 3. 步骤7中CA获得的属性值符合规范要求，测试成功。
备注	无

6.2.7 PropertyEnumerator 接口测试

测试编号	6.2.7
测试项目	属性枚举器测试
测试目的	验证属性枚举器枚举、重置、取属性的功能是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-9。 4. TA接收CA传入的属性类型（遍历所有属性类型）和枚举起始偏移位置。 5. TA获取步骤4中对应属性类型的属性枚举器对象。 6. TA依次遍历属性枚举器对象，并获得其中的所有属性对象。 7. TA重置属性枚举器对象。 8. TA从枚举起始偏移位置开始依次读取属性枚举器中的属性对象。 9. TA将步骤6、8的结果返回给CA。 10. CA关闭会话。 11. CA关闭上下文。
预期结果	1. 步骤1-3返回值符合规范要求，表示成功。 2. 步骤9中CA获得返回的结果与预期一致，测试成功。
备注	无

6.2.8 TEERuntimeException

测试编号	6.2.8
测试项目	异常捕获测试
测试目的	验证异常码的设置和抛出功能是否正确
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA抛出TEERuntimeException类异常码。 5. CA捕获异常码。 6. CA关闭会话。

	7. CA关闭上下文。
预期结果	1. 步骤1-3返回值符合规范要求，表示成功。 2. 步骤6中CA获得的异常码符合规范要求，测试成功。
备注	无

6.3 tee.trustedstorage 包

6.3.1 TEEObject 类测试

测试编号	6.3.1
测试项目	TEEObject 类测试
测试目的	验证获取对象的信息和限制属性使用的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3- 6验证获取对象信息功能 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA获取CA传入的算法类型。 5. TA根据步骤4中获取的算法类型创建并初始化TransientObject类对象tbo1。 6. TA获取tbo1的信息，包括类型、密钥、空间大小等。 7. TA将步骤6中数据返回给CA。 步骤8- 11验证限制对象的属性功能 8. CA发送指令到TA，触发TA执行步骤9-13。 9. TA获取CA传入的算法类型和限制类型（遍历可限制的属性值）。 10. TA根据步骤9中获取的算法类型创建并初始化TransientObject类对象tbo2。 11. TA根据步骤9中获取的限制类型限制tbo2的属性使用。 12. TA构造相应的算法类对象，并为其设置操作对象tbo2。 13. TA对算法类对象执行步骤11中被限制的操作，并将结果返回给CA。 14. CA关闭会话。 15. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤7中CA获取的信息符合规范对相应算法类型的要求，测试成功。 3. 步骤13中CA获取的错误码符合规范要求，测试成功。
备注	无

6.3.2 Attribute 类测试

测试编号	6.3.2
测试项目	Attribute 类测试
测试目的	验证获取指定标识属性、缓冲区属性及大小，以及对不同类型属性的取值和赋值功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。

	2. CA打开会话并与TA约定参数类型。 步骤3- 5验证属性对象的获取功能 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA获取CA传入的算法类型。 5. TA根据步骤4中的算法类型获得指定算法ID的attribute对象。 6. TA获取步骤4中attribute对象的ID和类型并将结果返回给CA。 步骤7- 13验证缓冲区属性及大小的获取、取值和赋值功能 7. CA发送指令到TA，触发TA执行步骤8-13。 8. TA获取CA传入的缓冲区类型数据a。 9. TA对步骤8获得的值a执行指定规则变换为新值并存入该attribute对象。 10. TA获取步骤9中attribute对象的值和大小，并将结果返回给CA。 11. CA关闭会话。 12. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤6中CA获得的值符合规范对相应算法类型的要求，测试成功。。 3. 步骤10中CA获得的值和预期结果一致，测试成功。
备注	无

6.3.3 TransientObject 类测试

测试编号	6.3.3
测试项目	TransientObject 类测试
测试目的	验证使用不同算法生成密钥、填充未初始化对象容器、复位 TransientObject 对象状态的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-7验证密钥生成和获取功能 3. CA发送指令到TA，触发TA执行步骤4-7。 4. TA获取CA传入的算法类型。 5. TA根据步骤4中不同算法类型创建TransientObject类对象（遍历所有算法）。 6. TA为该TransientObjetct对象生成密钥。 7. TA获取该TransientObjetct对象填充的密钥，并将结果返回给CA。 步骤8-13验证复位功能 8. CA再次发送指令到TA，触发TA执行步骤9。 9. TA再次初始化TransientObject对象。 10. CA获取TA的返回。 11. CA再次发送指令到TA，触发TA执行步骤12-13。 12. TA对该TransientObject对象执行复位操作。

	13. TA再次初始化TransientObject对象，并返回任意值给CA。 14. CA关闭会话。 15. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤7中CA获得的数据不为空并且长度与步骤4中设置的算法密钥长度一致，测试成功。 3. 步骤10中CA获取到的错误码符合规范，测试成功。 4. 步骤13中CA能够获取到TA返回的任意数据，测试成功。
备注	无

6.3.4 PersistentObject 类测试

测试编号	6.3.4
测试项目	PersistentObject 类测试
测试目的	验证创建、访问、删除持久化对象及读写数据的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-13验证创建、打开、关闭持久化对象，以及读写数据的功能 3. CA发送指令到TA，触发TA执行步骤4-13。 4. TA获取CA传入的需要进行持久化存储的数据和位置偏移值。 5. TA根据不同访问方式创建并删除持久化对象obj1（遍历所有访问方式）。 6. TA创建持久化对象obj1（访问方式为读/写/读元数据），根据步骤4中的位置偏移值设置写入位置，写入需要持久化存储的数据。 7. TA对持久化对象obj1执行关闭操作并返回。 8. CA获得TA返回值后关闭会话。 9. CA再次打开会话。 10. CA发送指令到TA，触发TA执行步骤11-13。 11. TA获取CA传入的位置偏移值。 12. TA以READ方法打开持久化对象obj1，根据步骤4中的位置偏移值设置读取位置，并读取对象数据。 13. TA将读取结果返回给CA。 步骤14-18验证持久化对象标识的重命名功能 14. CA发送指令到TA，触发TA执行步骤15-18。 15. CA向TA传入重命名的对象标识。 16. TA以WRITE_META方法打开持久化对象obj1。 17. TA根据步骤13获得的对象标识重命名obj1的对象标识。 18. TA获取obj1的对象标识并返回给CA。 步骤19-26验证持久化对象的关闭删除功能。 19. CA发送指令到TA，触发TA执行步骤20-22。

	20. CA向TA传入obj1的对象标识。 21. TA获得持久化对象obj1。 22. TA对持久化对象obj1执行关闭删除操作并返回。 23. CA发送指令到TA，触发TA执行步骤24-25。 24. CA向TA传入obj1的对象标识。 25. TA打开持久化对象obj1。 26. CA获得TA的返回值。 27. CA关闭会话。 28. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤9、10返回值为0x00000000，表示成功。 3. 步骤13中CA获得的数据与步骤4中从位置偏移值开始获取的传入数据一致，测试成功。 4. 步骤14返回值为0x00000000，表示成功。 5. 步骤18中CA获得的数据与步骤13中设置的标识一致，测试成功。 6. 步骤19返回值为0x00000000，表示成功。 7. 步骤26中CA获得的错误码符合规范要求，测试成功。
备注	无

6.3.5 PersistentObjectEnumerator 类测试

测试编号	6.3.5
测试项目	PersistentObjectEnumerator 类测试
测试目的	验证查找及获取持久化对象、对枚举器复位和指定可信存储空间标识的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-11。 4. TA获取CA传入的多个需要持久化的数据。 5. TA根据步骤4传入的数据依次创建并初始化多个持久化对象P1、P2、P3，并将其存储到指定存储空间。 6. TA获取指定存储空间的持久化枚举器对象。 7. TA判断该持久化枚举器对象是否有持久化对象。 8. TA对持久化枚举器进行枚举获取对象P1、P2、P3中存储的数据。 9. TA对持久化对象的枚举器复位。 10. TA再次对持久化枚举器进行枚举并获取对象中存储的数据。 11. TA将步骤7、8、10中的结果发送给CA。 12. CA关闭会话。 13. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤7中CA获取到的结果true，测试成功。 3. 步骤8中CA获取的结果与步骤4中CA设置的数据一致，测试成功。

	4. 步骤10中CA获得数据与步骤4中CA设置的数据一致，测试成功。
备注	无

6.3.6 异常信息类测试

测试编号	6.3.6
测试项目	异常信息测试
测试目的	验证使用指定的错误码抛出安全存储包拥有的异常类 StorageException 的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA抛出StorageException类的异常码。 5. CA捕获异常码。 6. CA关闭会话。 7. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤5中CA抛出异常码符合规范，测试成功。
备注	无

6.4 tee.cryptography 包

6.4.1 Operation 类测试

测试编号	6.4.1
测试项目	Operation 类测试
测试目的	验证获取、重置 Operation 对象信息及设置密钥的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-8 验证Operation对象的获取及密钥设置功能。 3. CA发送指令到TA，触发TA执行步骤4-8。 4. TA获取CA传入的算法类型和密钥数据。 5. TA为operation类对象做初始化准备，包括创建相应的临时对象和属性（遍历所有算法类型）。 6. TA根据步骤5中算法类型创建并初始化operation类对象。 7. TA 为该operation类对象放入密钥数据。 8. TA获取该operation类对象的信息。 步骤9-12 验证operation中密钥的重置功能. 9. TA根据步骤4中获取的算法类型初始化算法所需对象。 10. TA在执行多步骤算法的中间步骤中重置该operation类对象。 11. TA重新执行多步骤算法。

	12. TA将步骤8、11的结果数据返回给CA。 13. CA关闭会话。 14. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤12中CA获得TA返回的operation对象信息与传入的密钥数据相同，CA获得的TA返回的算法执行数据与预期结果一致，测试成功。
备注	无

6.4.2 AE 类测试

测试编号	6.4.2
测试项目	验证认证加解密算法测试
测试目的	验证 AE 类初始化、部分认证加密和全部认证加解密操作的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-11 验证认证加密算法和部分认证加密算法的功能 3. CA发送指令到TA，触发TA执行步骤4-11。 4. TA获取CA传入的需要进行认证加密的数据m和密钥。 5. TA为AE类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有认证加解密算法）。 6. TA创建AE类对象。 7. TA使用AE对象进行初始化认证加密操作。 8. TA使用AE对象对指定的全部明文m进行认证加密操作。 9. TA使用AE对象对部分明文m1进行认证加密操作。 10. TA使用AE对象对步骤10中的结果数据和剩下的明文m2进行加密操作。TA将步骤8、10的结果返回给CA。 步骤12-11 验证认证解密算法的功能 11. CA发送指令到TA，触发TA执行步骤13-18。 12. TA获取CA传入的需要进行认证解密的密文数据m和密钥。 13. TA为AE类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有认证加解密算法）。 14. TA创建AE类对象。 15. TA使用AE对象进行初始化认证解密操作。 16. TA使用AE对象对密文数据m进行认证解密操作。 17. TA将步骤17的结果返回给CA。 18. CA关闭会话。 19. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤11中CA获取到TA的返回值符合预期要求，测试成功。 3. 步骤12返回值为0x00000000，测试成功。

	4. 步骤18中CA获取到TA的返回值符合预期要求，测试成功。
备注	无

6.4.3 Cipher 类测试

测试编号	6.4.3
测试项目	加解密测试
测试目的	验证对称加解密和非对称加解密算法的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-11对称算法测试 3. CA发送指令到TA触发TA执行步骤3-11。 4. TA获取CA输入的需要进行加密或者解密的数据m和相关密钥。 5. TA为cipher类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有对称算法）。 6. TA创建Cipher对象。 7. TA使用Cipher对象初始化加解密操作。 8. TA使用Cipher对象对数据m进行加密或解密操作。 9. TA使用Cipher对象对部分数据m1进行加密或解密数据。 10. TA使用Cipher对象对步骤9中的结果数据和剩下的数据m2进行加密或解密操作。 11. TA将步骤8、10的结果返回给CA。 步骤12-17非对称算法测试 12. CA发送指令到TA触发TA执行步骤13-17。 13. TA获取CA输入的需要进行加密或者解密的数据m和相关密钥。 14. TA为cipher类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有非对称算法）。 15. TA创建Cipher对象。 16. TA使用Cipher对象对数据m进行加密或解密操作。 17. TA将步骤16的结果返回给CA。 18. CA关闭会话。 19. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，测试成功。 2. 步骤11中CA获取到TA的返回值符合预期要求，测试成功。 3. 步骤12返回值为0x00000000，测试成功。 4. 步骤17 中CA获取到TA的返回值符合预期要求，测试成功。
备注	无

6.4.4 KeyDerivation 类测试

测试编号	6.4.4
测试项目	KeyDerivation 类密钥分散方法测试

测试目的	验证设置操作对象和完成密钥分散操作的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA触发TA执行步骤4-11。 4. TA获取CA输入的需要进行分散操作的密钥。 5. TA为创建密钥分散操作类对象做初始化准备，包括创建和初始化相应的临时对象和属性（遍历所有分散算法）。 6. TA创建密钥分散操作类对象。 7. TA执行密钥分散操作，并将结果返回给CA。 8. CA关闭会话。 9. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤7中CA获得的返回结果符合规范要求，测试成功。
备注	无

6.4.5 MessageDigest 类测试

测试编号	6.4.5
测试项目	计算消息摘要及 Mac 值的方法测试
测试目的	验证消息摘要算法及 Mac 算法的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA触发TA执行步骤4-7。 4. TA获取CA传入的需要计算摘要或Mac值的数据m（如需进行Mac值计算还应传入密钥k）。 5. TA为创建MessageDigest类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有摘要及Mac算法）。 6. TA创建MessageDigest类对象。 7. TA使用MessageDigest对象对m完成消息摘要操作或Mac算法操作，生成消息摘要值或者Mac值并返回给CA。 8. CA关闭会话。 9. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功 2. 步骤7中CA获得的返回值符合规范要求，表示成功。
备注	无

6.4.6 Signature 类测试

测试编号	6.4.6
测试项目	签名及验签算法测试
测试目的	验证对消息的签名、验证签名及摘要签名算法的功能是否实现。
测试步骤	具体测试步骤如下：

	1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-9验证签名算法功能 3. CA发送指令到TA触发TA执行步骤4-9。 4. TA获取CA传入的需要进行签名的数据m和相关密钥。 5. TA为创建Signature类对象做初始化准备，包括创建相应的操作对象和属性（遍历所有签名算法）。 6. TA创建Signature类对象。 7. TA使用Signature对象进行初始化操作。 8. TA使用Signature对象对数据执行签名算法。 9. TA将结果返回给CA。 步骤10-14验证验签算法功能 10. CA发送指令到TA触发TA执行步骤11-14。 11. TA获取CA传入的需要进行验签的正确数据s1、错误数据s2和相关密钥。 12. TA使用Signature对象对正确数据s1执行验签算法。 13. TA使用Signature对象对错误数据s2执行验签算法。 14. TA返回步骤12、13的结果给CA。 15. CA关闭会话。 16. CA关闭上下文。
预期结果	1. 步骤1-3返回值为0x00000000，测试成功。 2. 步骤9、14中CA获取的返回值符合预期要求，测试成功。
备注	无

6.4.7 RandomData 类测试

测试编号	6.4.7
测试项目	RandomData 类测试
测试目的	验证生成随机数的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA获取CA传入的长度值。 5. TA创建RandomData 类对象。 6. TA根据步骤4中获得的长度值生成的相应长度的随机数（重复多次），并将所有结果返回给CA。 7. CA关闭会话。 8. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，测试成功。 2. 步骤6中CA获得的所有数据长度和步骤4指定长度一致，用工具测试其随机性，测试成功。

备注	需要借用随机数验证工具对生成的数据的随机性进行检测

6.4.8 异常信息类测试

测试编号	6.4.8
测试项目	异常信息测试
测试目的	验证抛出异常码是否正确。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-6。 4. TA抛出算法异常类的异常码。 5. CA获取TA抛出的异常码。 6. CA关闭会话。 7. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤5中CA获得的异常码符合规范，测试成功。
备注	无

6.5 tee.time 包

6.5.1 Time 类/TimeData 类测试

测试编号	6.5.1
测试项目	Time 类测试
测试目的	验证获取当前 REE 系统时间、当前系统时间、持久化时间及等待的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 步骤3-5验证当前REE系统时间功能 3. CA发送指令到TA，触发TA执行步骤4-5。 4. TA获取当前REE系统时间t1并返回给CA。 5. CA接收到TA的返回结果后，获取Android当前系统时间t2。 步骤6-8验证等待功能 6. CA发送指令到TA，触发TA执行步骤7-8，获取当前时间t3。 7. TA执行等待t4秒然后返回值。 8. CA收到返回值时获取当前时间t5，并计算t5-t3。 步骤9-16验证持久化时间功能 9. CA获取当前系统时间t6。 10. CA发送指令到TA，触发TA执行步骤11。 11. TA设置持久化时间并返回。 12. CA接收到返回后，关闭会话。

	13. CA再次打开与TA的会话。 14. CA再次发送指令到TA，触发TA执行步骤15。 15. TA获取持久化时间t7并将其返回给CA。 16. CA接收到返回结果后获取当前系统时间t8，并将t8-t6的差值和t7进行对比。 17. CA关闭会话。 18. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤5中CA获得的t2和步骤4中TA的返回结果t1的误差符合要求，表示成功。 3. 步骤8中CA计算t5-t3的结果和t4的误差符合要求，表示成功。 4. 步骤10、12、13、14返回值为0x00000000，表示成功。 5. 步骤16中CA返回t8-t6的值与t7的误差符合要求，表示成功。
备注	无

6.6 tee.util 包

6.6.1 Util 类测试

测试编号	6.6.1
测试项目	Util 类测试
测试目的	验证比较、复制、填充及连接数组内容的功能是否实现。
测试步骤	具体测试步骤如下： 1. CA初始化上下文。 2. CA打开会话并与TA约定参数类型。 3. CA发送指令到TA，触发TA执行步骤4-9。 4. TA接收CA传入的两个不同数组str1和str2、以及填充偏移量。 5. TA获取数组str1和str2的值。 6. TA对两个数组进行比较。 7. TA复制str1的内容到str2。 8. TA在数组str1中从步骤4中获得的填充偏移量开始填充数组str2的值。 9. TA连接两个数组str1和str2。 10. TA将步骤5、6、7、8、9返回值发送给CA。 11. CA关闭会话。 12. CA关闭上下文。
预期结果	1. 步骤1~3返回值为0x00000000，表示成功。 2. 步骤10中CA获得的TA返回的结果与预期一致，测试成功。
备注	无

附 录 A
（资料性附录）
文档编写记录

修订时间	修订后版本号	修订内容
2017.4.1	V1.0	全文格式