

移动智能终端补充设备标识体系 统一调用 SDK

常见问题问答 F&Q

编写人	移动安全联盟
文档版本	V2. 5
最新修订日期	2021 年 7 月 19 日

权利声明

任何单位或个人使用移动智能终端补充设备标识体系统一调用 SDK 前，应当仔细阅读本权利声明并确认同意本权利声明所述内容，否则不得使用本 SDK。您理解并承诺，您使用 SDK 的行为，即视为您已了解并完全同意本权利声明的各项内容，且您将基于这些声明承担相应的法律责任与义务。

1、移动智能终端补充设备标识体系统一调用 SDK 由中国信息通信研究院泰尔终端实验室、移动安全联盟整合提供，知识产权归中国信息通信研究院所有。任何单位或个人未经授权不得修改、复制、发行、出租、传播或翻译 SDK，不得逆向、破解、篡改、二次打包、公开、贩卖 SDK，不得向任何第三方披露 SDK，不得进行二次分发，不得利用 SDK 直接获利或用于其他商业用途，如基于本 SDK 向第三方提供与本 SDK 相似功能。如有违反中国信息通信研究院保留追究其法律责任的权利。

2、移动智能终端补充设备标识体系依据电信终端产业协会（TAF）、移动安全联盟（MSA）联合推出的团体标准《移动智能终端补充设备标识规范》开发，移动智能终端补充设备标识体系统一调用 SDK 集成设备厂商提供的接口，并已获得相应设备厂商的授权。

3、任何单位或个人使用本 SDK 获取相关标识数据，并欲据此标识数据进行使用、分析、交易或其他行为前，应慎重考虑这些行为的合法性、合规性、真实性和安全性等，并采取相应措施。如因未采取措施所导致的信息泄露、权利侵犯、财产毁损、人身伤亡等违法违规行为及因该等行为所造成的损害后果，与中国信息通信研究无关，中国信息通信研究亦不承担任何直接、间接、附带或衍生的损失和责任。

4、移动智能终端补充设备标识体系统一调用 SDK 由中国信息通信研究院泰尔终端实验室、移动安全联盟共同负责 SDK 的合规管理和后期维护，移动安全联盟官方网站（<http://www.msa-alliance.cn/col.jsp?id=120>）与官方邮箱（msa@caict.ac.cn）是目前唯一合法发布本 SDK 版本、代码、文档的渠道，任何从其他个人、企业或组织渠道获取的 SDK 均与中国信息通信研究院泰尔终端实验室、移动安全联盟无关，请仔细甄别。

中国信息通信研究院

2020 年 3 月 17 日

一. F&Q

1、 什么是移动智能终端补充设备标识体系统一调用 SDK

中国信息通信研究院泰尔终端实验室、移动安全联盟集成开发的能支持全终端厂商的统一调用 SDK，分发给移动互联网应用，协助移动应用开发者更便捷的访问移动智能终端补充设备标识体系，推进相关业务。

2、 SDK 实现功能

SDK 整合不同终端厂商补充设备标识体系调用接口，移动互联网应用通过 SDK 可直接获取所需标识，不用针对不同终端品牌与型号进行适配。

3、 SDK 引入的必要性问题

SDK 接入是自愿的，SDK 为获取本体系 ID（OAID、VAID、AAID）的官方的最便捷的方案，但若企业评估后认为不需获取 ID，也可不接入 SDK。

4、 SDK 支持终端范围

厂商名称	支持版本
华为	HMS 2.6.2 及以上
小米	MIUI 10.2 及以上版本
vivo	Android 9 及以上版本
OPPO	colorOS 6 大部分覆盖，colorOS 7 及以上全覆盖
联想	ZUI 11.4 及以上版本
三星	Android 10 版本
魅族	Android 10 版本
努比亚	Android 10 版本
中兴	Android 10 版本
华硕	Android 10 版本
一加	Android 10 版本
黑鲨	Android 10 版本
摩托罗拉	Android 10 版本
Freeme OS	Android 10 版本
酷赛（铂睿智恒）	Android 10 版本
Realme	colorOS 6 大部分覆盖，colorOS 7 及以上全覆盖
荣耀	Android 10 版本

厂商名称	支持版本
酷派	Android 10 版本

5、 不支持的设备怎么办

不支持的终端设备或终端版本无法获取 ID，移动安全联盟将持续推动终端设备支持本标识体系。

6、 非 APP 类产品如 SDK 等，集成“统一调用 SDK”问题

非 APP 类产品接入，可能会存在 BUG（例如当一个 SDK 接入本 SDK，但是这个 SDK 的宿主 APP 也接入本 SDK，那在 APP 调用时就可能会产生冲突），同时数据安全与个人信息保护问题不可控，故原则上不提供非 APP 类产品接入通用 SDK，需要定制化开发才能正常运行。如有意向获取定制化开发版本，可以向 msa@caict.ac.cn 发送邮件。

根据权利声明，任何单位或个人未经授权不得修改、复制、发行、出租、传播或翻译 SDK，不得逆向、破解、篡改、二次打包、公开、贩卖 SDK，不得向任何第三方披露 SDK，不得进行二次分发，不得利用 SDK 直接获利或用于其他商业用途，如基于本 SDK 向第三方提供与本 SDK 相似功能。

7、 四个 ID 区别，SDK 获取情况

设备唯一标识符（UDID）：设备唯一硬件标识，设备生产时根据特定的硬件信息生成，可用于设备的生产环境及合法性校验。**不对第三方应用提供获取接口，无法通过 SDK 获取。**

匿名设备标识符（OAID）：可以连接所有应用数据的标识符，移动智能终端系统首次启动后立即生成，可用于广告业务。可以通过 SDK 获取到接口状态（重置、关闭）、ID 值。

开发者匿名设备标识符（VAID）：用于开放给开发者的设备标识符，可在应用安装时产生，可用于同一开发者不同应用之间的推荐。可以通过 SDK 获取到 ID 值。

应用匿名设备标识符（AAID）：第三方应用获取的匿名设备标识，可在应用安装时产生，可用于用户统计等。可以通过 SDK 获取到 ID 值。

少数终端企业可能只支持部分 ID 的获取。

8、 四个 ID 的长度与格式

四个 ID 最长为 64byte，位数没有固定要求。

OAID 格式示例：

华为：1fe9a970-efbb-29e0-0bdd-f5dbbf751ab5

小米：e6ee0f4b6b67cf8b

vivo：9c293b57bdf31a7271cc48113dc0c217249aca2dcd98c8c5a55d5861002197c0

OPPO：24D7654BA68343DB8E81DD16697CAA5050d1e0e769d113a84794f6e8fceececa5b

9、 移动智能终端补充设备标识体系与 SDK 隐私合规问题

移动智能终端补充设备标识体系与 SDK 严格遵守我国《网络安全法》、《数据安全法》和《电信和互联网用户个人信息保护规定》等相关法律法规和《信息安全技术 个人信息安全规范》等国家标准要求。SDK 不会主动收集任何数据，只在 APP 调用时获取以下数据，用于本地判断，**不会进行任何网络传输**：

- 设备制造商、设备型号、设备品牌，用于判断终端调用接口
- 设备网络运营商名称，用于判断虚拟机环境
- APP 包名，用于校验签名

如开发者有需求，可签订保密协议，保障双方的数据安全与合作保密性。

开发者请自行评估使用本 SDK 可能涉及到的法律风险，并制定使用本 SDK 获取 ID 时的隐私政策。

10、udid 为什么获取不到

udid 不对第三方应用提供获取接口，无法通过 SDK 获取。

11、oaid 重置/关闭问题

oaid 在恢复出厂设置后会重置，并提供给用户关闭和重置的权利，重置后 oaid 值会改变。除这两种情况外，根据标准说明“匿名设备标识符自身可定期重置”，表示终端在特定情况下保留重置 oaid 值的能力，如根据监管需要等。

oaid 关闭后情况，不同终端企业逻辑不同，一种情况是关闭后不可获取 oaid（返回 null 或全 0），另一种情况是关闭后告知已关闭（可通过 isLimited 接口获取）并可获取重置后的 oaid（此种显示的是禁止跟踪等说法），当前两种方法都均存在，但第二种风险较大不排除后续会有限制的可能。同时，如果选用第二种方案要求开发者不可将获取到的 ID 用于追踪用户，如果发现违反会有相关处理。

12、void 如何判断为同一开发者

void 在同一终端同一开发者情况下，不同应用获取到的值相同，不同终端企业判断是否为同一开发者方式不同，一种是通过 appid 判断，相同 appid 获取到的 void 相同，此种方式需要开发者在配置文件中填写对应终端应用商店的 appid，如 vivo 品牌；一种是通过 app 签名信息判断，相同 app 签名信息获取到的 void 相同，此种情况不需填写相应 appid。

13、appid 如何获取

appid 需要移动互联网应用开发者根据应用使用需求到不同终端厂商的应用商城申请，具体需咨询相关厂商，不需要填写其他第三方应用商店的 appid。appid 只与 VOID 的获取有关，用于判断是否为同一开发者，如不需获取 void 可不填写。目前只需设置 vivo 的 appid。

14、SDK 使用是否需要联网

SDK 是否需要联网跟设备类型有关，多数设备厂商都需要联网，特别是获取 void、aaid 需要去厂商后台检验计算获取，联网时间发生在调用接口的时候。

15、SDK 是否支持模拟器

移动智能终端补充设备标识必须是真机才有效，而且是已经支持的真机，支

移动安全联盟

（<http://www.msa-alliance.cn>）

持列表见文档，虚拟机没办法获得移动智能终端补充设备标识。

16、 supplierconfig.json 配置文件如何修改？

supplierconfig.json 文件内容无需修改和补充，只需填写对应 appid，并放到 assets 目录下即可。

17、 证书有什么用

SDK 1.0.26 版本引入证书校验机制，会对 APP 包名进行校验，每个 APP 包名对应一个证书文件（应用包名.cert.pem），只有包名校验成功的 APP 才能获取 ID。

18、 如何申请证书

证书需要填写 example_batch.csv 后发送到 msa@caict.ac.cn 申请，注意每个包名对应一个签名，申请时要将所需的全部包名填写到表格中。默认证书有效期一年，快过期时请及时申请更新。

19、 出现证书初始化失败怎么办

请先按以下条目排查：

- 1) 真机系统时间（日期）是否正确；
- 2) 传入的证书字符串是否包含了证书文件的所有字符内容，包括换行符；
- 3) 证书和 APP 包名是否一一对应，不能混用；
- 4) 若用于插桩测试（Instrumentation API），则需确保传入的 context 为 TargetContext；
- 5) 在提供的 demo 中使用下发的证书，会出现证书初始化失败。demo 的包名和申请的证书中的包名不一致，无法通过证书校验，请在自己的 APP 中使用申请的证书。

20、 华为手机 oaid 返回 00000000-0000-0000-0000-000000000000

华用手机在用户启用“限制广告跟踪”开关后，oaid 将会重置为 00000000-0000-0000-0000-000000000000。

21、 小米手机异常，java.lang.NoSuchMethodException: getDefaultUDID [class android.content.Context]

这个只是输出的一个日志，关闭日志，自然就没有了，不会引起系统崩溃，放心使用。

22、 出现“dynamic section has invalid link(0) sh_type: 0 (expected SHT_STRTAB)”崩溃

这个问题是符号被 strip 掉了，对比一下提供的 aar 中的 so 大小与生成的 APK 中 so 的大小是否一致，不一致就是符号被处理了，加固后的 so 无法识别启动不了，也就是说 doNotStrip 没有设置好，需要在 app 的 gradle 里设置。

23、 报错提示 “No implementation found for boolean com.bun.miitmdid.core.MdidSdkHelper.InitCert”

这个问题是没有加载安全库导致的，参考 DemoHelper，在调用前需载入 SDK

移动安全联盟
(<http://www.msa-alliance.cn>)

安全库 “System.loadLibrary(“nllvml623827671”);”。

24、三星 10 的手机上会产生崩溃，主要是两个原因

(1) 1.0.13 版本加固后，集成到 app 过程中，混淆的时候，会把一些类优化掉，导致运行时找不到某些类定义。解决添加混淆选项

#msa sdk 混淆配置

```
-keep class com.bun.** {*;}
-keep class com.asus.msa.** {*;}
-keep class com.heytap.openid.** {*;}
-keep class com.huawei.android.hms.pps.** {*;}
-keep class com.meizu.flyme.openidsdk.** {*;}
-keep class com.samsung.android.deviceidservice.** {*;}
-keep class com.zui.** {*;}
-keep class com.huawei.hms.ads.** {*; }
-keep interface com.huawei.hms.ads.** {*; }
-keepattributes *Annotation*
-keep @android.support.annotation.Keep class **{
    @android.support.annotation.Keep <fields>;
    @android.support.annotation.Keep <methods>;
}
```

(2) 最早 1.0.10 版本提供了一个 helper 类，里边有直接调用的相关代码，因为这个和加固是冲突的，所以应该删掉或注释掉。

解决办法：把如下代码删掉或注释掉。

```
private int DirectCall(Context context) {
    MdidSdk sdk = new MdidSdk();
    return sdk.InitSdk(context, this);
}
```

25、其他崩溃解决办法

描述问题，把 SDK 版本号，崩溃手机型号，崩溃手机版本号，APP 是否加固、是否有热更新等收集一下，然后加上 apk，打包发到 msa@caict.ac.cn。

二、联系方式

其他任何疑问可发送邮件至 msa@caict.ac.cn，注明公司、联系人、问题详情。