

1.1 禁止应用读取 Device ID

1.1.1 背景介绍

Q版本将限制应用访问不可重设的设备识别码，如 IMEI、序列号等，所有获取设备识别码的接口都增加了新的权限：`READ_PRIVILEGED_PHONE_STATE`，该权限需要系统签名的应用才能申请，意味着三方应用无法获取设备识别码。

Device Identifiers

Restricting access to persistent device identifiers

Immutable (permanent, non-resettable) device identifiers are sensitive, when collected.

In Q:

Immutable device identifiers will be available only via a new permission `READ_PHONE_STATE_PRIVILEGED`. This permission is restricted to preloaded apps.

- Telephony: IMEI, MEID, ESN, IMSI numbers
- Build, SIM, or USB serial numbers

1.1.2 兼容性影响

1. 影响范围

90%+的应用都通过申请老的`READ_PHONE_STATE`权限获取device id，这些应用都需要适配该变更

2. 兼容性问题

- `TargetSdkVersion < Q` 并且没有申请`READ_PHONE_STATE`权限，或者 `TargetSdkVersion >= Q`，获取device id会抛异常`SecurityException`
- `TargetSdkVersion < Q` 并且申请了`READ_PHONE_STATE`，通过`getDeviceId`接口读取的值为`Null`
- 无法获取到device id，会对应用依赖device id的功能产生影响

3. 问题和场景

数据统计、推荐、用户历史数据记录、广告、用户画像等等

1.1.3 适配指导

1. 谷歌提供的适配指导文档

唯一标识符最佳做法: <https://developer.android.google.cn/training/articles/user-data-ids>

官方文档: <https://developer.android.google.cn/preview/privacy/data-identifiers>

2. 开发者可以尝试使用 Android ID 替换 Device ID

1. 获取代码:

```
Settings.System.getString(context.getContentResolver(), Settings.Secure.ANDROID_ID);
```

2. Android ID和Device ID的区别:

- a) 手机恢复出厂设置, Android ID可以重置, 但是Device ID无法重置
- b) 对于安装在运行 Android 8.0 的设备上的应用, ANDROID_ID 的值现在将根据应用签署密钥和用户确定作用域。应用签署密钥、用户和设备的每个组合都具有唯一的 ANDROID_ID 值。因此, 在相同设备上运行但具有不同签署密钥的应用将不会再看到相同的 Android ID (即使对于同一用户来说, 也是如此), 具体可参考O版本的变更: <https://developer.android.com/about/versions/oreo/android-8.0-changes#privacy-all>

3. 接入华为提供的 ID

目前华为的OAID和ODID接口还在开发中, 开发完成之后再补充适配信息:

1. OAID(Open Anonymous ID): 开放匿名ID, 只能用于广告场景(禁止使用在其他场景), 用户可以在手机上进行关闭和重置Open Anonymous ID, 重置之后OAID值会变化。OAID采用uuid作为OAID的值, 第三方开发者也可以获取。接入指导:

a) 接口文档:



PPS-Service接口
说明书_OAID.PDF

- b) 需要到华为应用市场下载最新的华为移动服务apk进行测试:



2. ODID(Open Device ID): 开放的设备标识，用于开放给开发者的设备标识，同一个开发者不同应用获取设备标识是相同的。

a) Open Device ID取值规则：

同一设备上运行的同一个供应商的应用程序，取值相同；

同一设备上不同供应商的应用程序，取值不同；

不同设备上同一个供应商的应用程序，取值不同；

不同设备上不同供应商的应用程序，取值不同；

b) 重置规则：

手机恢复出厂设置，所有供应商的取值全部重置；

同一设备上同一供应商的应用全部卸载后重新安装时，对该供应商的取值重置；

4. 老的 READ_PHONE_STATE 权限整改

由于获取device id权限的变更，通过申请老的READ_PHONE_STATE已经无法获取device id信息，所以如果应用申请敏感权限READ_PHONE_STATE的目的仅仅是为了获取device id，建议代码中增加版本判断，判断如果仅当运行的手机版本是Q以前的版本，才申请用户动态授权，否则申请的权限没有用，反而对用户产生打扰，具体代码参考：

```
if (Build.VERSION.SDK_INT <= Build.VERSION_CODES.P) {
    if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.READ_PHONE_STATE)
```

```
    != PackageManager.PERMISSION_GRANTED) {  
    Log.e(TAG, "not have READ_PHONE_STATE");  
    ActivityCompat.requestPermissions(this,  
        new String[]{Manifest.permission.READ_PHONE_STATE},  
        MY_PERMISSIONS_REQUEST_READ_PHONE_STATE);  
    } else {  
        Log.e(TAG, "have READ_PHONE_STATE");  
    }  
}
```