**1. Restrictions on non-SDK interfaces**

隐藏API的调用（即应用程序引用非SDK接口，无论采用直接、反射、JNI获取等手段都将受到限制）

名单分类

①Light grey list: targetSDK>=P时，警告；

②Dark grey list: targetSDK<P时，警告；>=p时，不允许调用；

③Black list:三方应用不允许调用；

适配方案：

（1）通过日志，找出应用调用的所有non-SDK接口名单，关键日志：

AccessinghiddenfieldLandroid/os/Message;->flags: I(lightgreylist, JNI)AccessinghiddenmethodLandroid/app/ActivityThread;->currentActivityThread()Landroid/app/ActivityThread;(darkgreylist, reflection)

（2）对于已经被禁止调用的接口需要整改，寻找替换该non-SDK接口的方案；

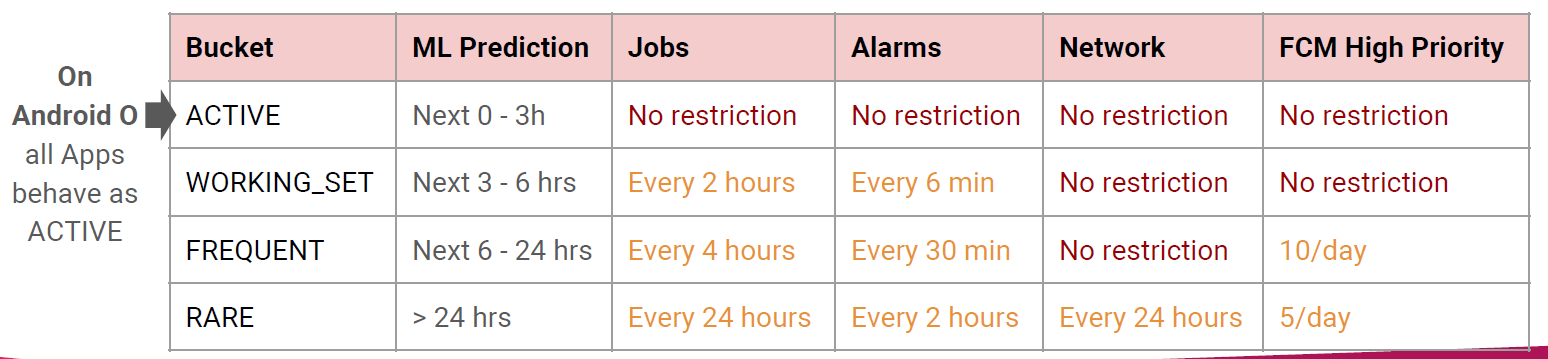
（3）如果必须要调用的non-SDK接口可以向谷歌申诉把接口加入到灰名单

<https://developer.android.google.cn/preview/restrictions-non-sdk-interfaces>

**2. Battery Improvements**

谷歌在P版本之前没有一个完整的功耗解决方案，OEM厂商分别开发各自的功耗方案，管控手段都包括了清理应用，功耗得到优化，但是同时也影响了三方应用的一些功能正常使用，谷歌为了解决这个问题在P版本提出了自己的功耗解决方案，该方案主要包含：Auto Awesome Battery（AAB），Extreme Battery Saver（EBS），Smart screen brightness。

（1）Auto Awesome Battery中，通过ML算法将应用进行分类，分别分为Active，Working set，Frequent，Rare。P之前基本都遵循Active的策略，但是P版本分类后对应的管控策略如下图：



（2）Extreme Battery Saver（EBS）谷歌超级省电模式；

（3）Smart screen brightness：屏幕亮度调节优化算法。

谷歌功耗方案对三方应用各种管控，存在导致应用后台功能无法正常使用的可能，特别是：IM、邮箱、闹钟、音乐（直播）、地图导航、运动健康、下载、日历等应用影响比较大。目前通过谷歌提供的调试命令验证：所有的应用都有可能会被分到管控的类型，对三方的后台功能是有影响的。

适配方案：对于以下后台功能，可能存在影响，如何判断是不是谷歌AAB方案导致的问题？谷歌提供了调试命令，可以获取应用所处的管控分类，并且可以让某个应用进入指定的分类进行管控，测试验证对应管控策略下的应用行为。

（1）Unplug (or adb shell dumpsys battery unplug )

（2）adb shell am get-standby-bucket <package name> ; //获取某个指定应用分类数据（5-50）

         5 EXEMPT/WHITELISTED

         10 ACTIVE

         20 WORKING\_SET

         30 FREQUENT

         40 RARE

         50 NEVER

（3） adb shell am set-standby-bucket <package name> <bucket>//强制把某个应用设置到指定的类别

（4）Programmatic: UsageStatsManager::getAppStandbyBucket()

<https://developer.android.google.cn/preview/features/power-details>

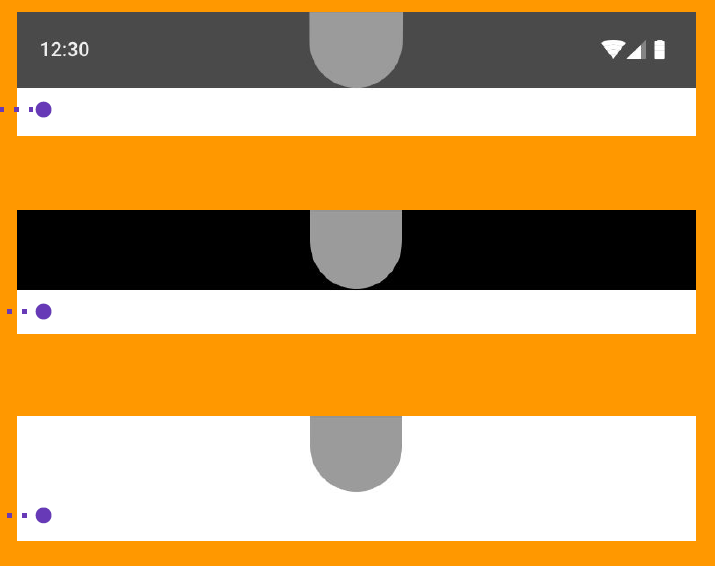
**3. 刘海屏适配（之前有调研过）**

(1）谷歌P版本提供了统一的挖孔屏方案和三方适配挖孔屏方案：

①对于有状态栏的页面，不会受到挖孔屏特性的影响；

②全屏显示的页面，系统挖孔屏方案会对应用界面做下移避开挖孔区显示；

③已经适配的P的应用的全屏页面可以通过谷歌提供的适配方案使用挖孔区，真正做到全屏显示。



（2）P版本三方适配挖孔屏方案：

①新增挖孔屏挖孔尺寸和位置接口

class WindowInsets { DisplayCutout getDisplayCutout(); } class DisplayCutout { int getSafeInsetLeft(); int getSafeInsetTop(); int getSafeInsetRight(); int getSafeInsetBottom(); Region getBounds(); }

②新窗口布局模式，允许应用程序请求是否在挖孔区域布局：

class WindowManager.LayoutParams { int layoutInDisplayCutoutMode; final int LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_DEFAULT; final int LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_ALWAYS; final int LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_NEVER; }

layoutInDisplayCutoutMode值说明：

a）LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_DEFAULT：默认情况下，全屏窗口不会使用到挖孔区域，非全屏窗口可正常使用挖孔区域。

b）LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_ALWAYS：窗口声明使用挖孔区域

c）LAYOUT\_IN\_DISPLAY\_CUTOUT\_MODE\_NEVER：窗口声明不使用挖孔区域

**（3）**适配指导

（1）在非挖孔屏P版本手机可以开启模拟挖孔屏调试的功能：

①在开发人员选项屏幕中，向下滚动到绘图部分，然后点击“模拟具有凹口的显示屏”设置项；

②选择挖孔尺寸信息；

（2）在挖孔屏调试打开之后，浏览应用的所有页面，测试所有遮挡问题，或者是下移导致的问题，对有问题的页面进行布局适配。，有状态栏的页面基本无需做特别处理，只需让状态栏和页面看起来比较优雅即可。全屏模式下，先开启刘海屏显示模式，在开启的情况下，获取刘海屏的宽度，然后再代码中将整个layout下移刘海的高度，或者保证应用的显示不被刘海遮挡。（全屏页和沉浸式标题栏要注意不要遮挡内容）

**4.  Camera、sensor和麦克风后台管控**

Android P 通过限制后台应用访问用户输入和传感器数据的能力增强了隐私性。增加Camera、sensor和麦克风的background限制，限制后台应用或服务的访问Camera或者麦克风，对前台应用无影响；

应用处于idle和gone的状态就会被管控，切换到后台一分钟应用就会进入idle状态，但是有前台服务的不会进入idle状态；

适配方案：需要后台访问麦克风、sensor和camera的时候增加前台服务。

**5. FLAG\_ACTIVITY\_NEW\_TASK被强制要求**

在P版本，如果不在Intent添加FLAG\_ACTIVITY\_NEW\_TASK，将无法通过非Activity的Context启动一个Activity，并且会抛异常。

比如在Service中启动Activity，如果Intent不添加FLAG\_ACTIVITY\_NEW\_TASK，就会抛异常

**6. 应用禁止访问xt\_qtaguid目录的文件**

对应用程序不再允许直接读取/ proc / net / xt\_qtaguid文件夹中的文件。原因是为了确保和运行Android P的设备保持一致，这些设备根本没有这些文件。

依赖这些文件的公共API，TrafficStats和NetworkStatsManager继续按预期工作。但是，不支持的cutils函数（如qtaguid\_tagSocket()）可能无法按预期方式工作，或者根本无法在不同的设备上工作。

**7. 前台服务权限**

应用如果想使用前台服务需要申请FOREGROUND\_SERVICE权限，这个权限是普通权限，如果不申请权限就直接启动前台服务会抛SecurityException。

**8. 不允许共享WebView数据目录**

应用程序不能再跨进程共享单个WebView数据目录。如果您的应用有多个使用WebView，CookieManager或android.webkit包中的其他API的进程，则当第二个进程调用WebView方法时，您的应用将崩溃。

该特性只影响已经适配P的应用，也就是targetSDK Version>=P。

**9. 权限相关修改**

CALL\_LOG， READ\_CALL\_LOG， WRITE\_CALL\_LOG和 PROCESS\_OUTGOING\_CALLS 从权限PHONE移动到新的 CALL\_LOG权限组。当查看电话相关信息时，必须申请CALL\_LOG权限组，否则会报SecurityException异常

Wi-Fi location访问限制，当进行Wi-Fi扫描时，必须具有ACCESS\_COARSE\_LOCATION 权限或ACCESS\_FINE\_LOCATION权限。否则无法获取扫描结果。

**10. Crypto provider在P版本去掉了**

从P版本开始，Crypto JCA provider被去掉了，调用SecureRandom.getInstance(“SHA1PRNG”, “Crypto”) 将会报NoSuchProviderException。

**11. 去除通过Build.serial获取设备识别码**

应用需要验证手机序列号必须要申请READ\_PHONE\_STATE权限，然后通过P版本新增的接口Build.getSerial() 来获取：

（1）通过Build.SERIAL获取不到真实数据：Build.serial:unknown

（2）通过Build.getSerial()才能获取真实的数据，需要用户授权READ\_PHONE\_STATE权限

该特性只影响已经适配P的应用，也就是targetSDK Version>=P。

参考文档：

<https://devcenter.huawei.com/consumer/cn/devservice/doc/50115>

<https://developer.android.google.cn/preview/behavior-changes>

Android P的新功能：

**1. 利用 Wi-Fi RTT 进行室内定位**

Android P 添加了对 IEEE 802.11mc Wi-Fi 协议的平台支持，应用可以使用全新的 [RTT API](https://developer.android.google.cn/reference/android/net/wifi/rtt/package-summary.html) 来测量与附近支持 RTT 的 Wi-Fi 接入点 (AP) 的距离。 设备必须已启用定位并开启了 Wi-Fi 扫描（在 Settings > Location 下），同时您的应用必须具有 ACCESS\_FINE\_LOCATION 权限。

通过这种精确性，您可以打造新的体验，例如楼内导航、基于精细位置的服务。

**2. 多摄像头支持和摄像头更新**

在配备双前置摄像头或双后置摄像头的设备上，您可以创建只配备单摄像头的设备所不可能实现的创新功能，例如无缝缩放、虚化和立体成像。 通过此 API，您还可以调用逻辑或融合的摄像头视频流，该视频流可在两个或更多摄像头之间自动切换。

**3. ImageDecoder**

一个的可以将PNG, JPEG, WEBP, GIF, or HEIF 格式的图片的转换成Drawable 或者Bitmap 对象的类。 可以使用 ImageDecoder 来代替BitmapFactory 和 BitmapFactory.Options解码图像。

**4. AnimatedImageDrawable**

Android P 引入了一个新的 AnimatedImageDrawable 类，用于绘制和显示 GIF 和 WebP 动画图像。

**5. JobScheduler 的优化**

Android P 中对 JobScheduler 进行了改进，使其可以更好地为用户处理网络相关的Job。

setEstimatedNetworkBytes()、setIsPrefetch() 和 setRequiredNetwork()

Job可以声明其预估的数据大小、信号预提取，并指定具体的网络要求，而运营商可以报告网络拥塞或无限流量。 然后，JobScheduler 根据网络状态管理工作。 例如，当网络拥塞时，JobScheduler 可能会延迟较大的网络请求。 如果使用的是无限流量网络，则 JobScheduler 可运行预提取作业以提升用户体验（例如预提取标题）。

**6. 安全增强功能**

Android P 引入了若干可提升应用和运行应用的设备安全性的行为变更。

比如传输层安全协议 (TLS) 实现变更，严格的 Seccomp 过滤器，对 ChaCha20 流式传输加密的支持等。

统一的指纹身份验证对话框，使用FingerprintDialog代替 FingerprintManager来显示指纹身份验证对话框。

 Android P 新增了Protected Confirmation API，这个新增的 API，应用可以利用 ConfirmationDialog 的实例向用户显示提示，请他们批准一个简短的声明。 应用可以通过这个声明再次确认，用户确实想完成一项敏感事务，例如付款。