[SCA] Crash 安全应急方案 - 调研文档

Ownership

Product Manager	*The PIC of Product manager*
Project Manager	*The PIC of Project Manager*
Native Dev	*The PIC of Android or iOS Dev*
RN Dev	*The PIC of React native dev*
Server Dev	*The PIC of non-native dev*

Resource

PRD	*The link of PRD(s), created by product manager*
Dependent Service Doc(Optional)	*SDK integration doc, official guideline, etc.*

Overview

对于Android应用来说，运行过程中发生错误则会发生崩溃（Crash，即抛出未处理的异常），系统的默认处理是关闭或重启应用。这对于用户来说是一个相当糟糕的体验，如果启动过程中崩溃甚至会导致应用不可用。预防崩溃是开发工作的重中之重，但一旦崩溃发生，我们还需要一套应急方案，特别是对于启动崩溃，我们希望也能保证用户可以一定程度上使用我们的应用。

Feasibility

- **Question 1 ： 是否支持全局崩溃后处理？**

  Answer： 结论是可以做自己定制的处理。

应用处理Java层异常靠的是Thread中的一个默认的UncaughtExceptionHandler，在ART初始化时（RuntimeInit.commonInit）就会设置KillApplicationHandler（实现了UncaughtExceptionHandler接口）。在发生异常时，会把异常信息告知AMS，AMS会把Log写入系统，并根据应用崩溃次数弹出系统的崩溃弹框，最终都会退出应用并终止虚拟机，同时根据用户操作决定是否重启应用。系统提供了setDefaultUncaughtExceptionHandler方法，支持插入或替换处理异常的逻辑。Bugly和Firebase这些监测应用崩溃日志的框架就是利用了这个Handler，插入了日志上报逻辑。

Native崩溃是基于linux信号量处理的，可以用Signal库的sigaction函数注册信号处理。

如果有必要的话，可以考虑加入ANR的后续处理。ANR是基于延时消息实现的，四大组件区分不同的实现，不便于统一监控。再往底层看，ANR也是会发出特定的信号量，可以考虑处理相关的信号量来处理ANR；也可以参考xCrash，利用FileObserver来观察/data/anr/下的trace文件，有新的trace文件内容即可判断发生了ANR。

Solution Overview

功能的实现包含两个方面：崩溃监控，以及后续安全的应急处理。

崩溃监控必须保证准确且及时地发现全部的崩溃，精准地判断崩溃的各种条件和信息，后续才能根据情况执行正确的应急处理。

Crash后的处理，可以考虑以下几个方案：

1. 偶现或不影响使用的问题，只需要**重启应用**。

2. 开启全局兜底逻辑，应用进入**兜底降级处理**。可以设置一个全局的兜底开关，开启兜底后，应用需要运行在一个安全稳定的代码中。为了保证业务尽可能不受影响，可以把一个稳定的版本作为兜底，后续的改动都运行在兜底之外，兜底版本也要不断迭代。
3. 如果兜底版本仍然有问题，可以尝试**清除应用数据**，再重启应用。
4. 如果还没能解决问题，就要考虑**热修复**。可以拉取特定的补丁包或全量包，操作安装修复来解决问题。

具体实现时，必须考虑如下方面：

1. 上述处理流程在应用崩溃后能正常运作。考虑到有OOM的问题，在主进程运行处理流程是不可靠的。所以应该在hold住主进程的基础上，开启一个安全进程来处理后续。
2. 为了能准确评估问题的严重程度，且考虑到与用户无法顺畅沟通的情况，在把问题上报到三方平台（Firebase或Bugly）的同时，还需要考虑增加一个离线模式（网络问题或三方平台未初始化导致无法把问题上报到三方平台）。如在界面上直接可选择展示问题信息，或者写入特定的本地文件。
3. 处理方案可供用户选择，且操作尽可能简易。在系统弹框弹出之前可以弹出自定义的选择弹框，可以选择去掉系统弹框。

崩溃捕获及应急处理的开源框架：bugsnag-android，xCrash, Cockroach，CustomActivityOnCrash。

	简介	崩溃类型	后续处理	开启安全进程
Bugsnag	崩溃上报，专门处理了启动崩溃	Java Native ANR	写入本地文件，上报服务端	是
xCrash	主要用于捕获崩溃	Java Native ANR	写入本地文件	否
Cockroach	自动尝试恢复Activity栈	Java	接口回调	否
CustomActivityOnCrash	崩溃后自定义页面展示	Java	自定义界面展示	是

这些框架任何一个都不能完全满足我们的需求，但崩溃捕获和后续处理都可以参考其实现。

Java层的崩溃，都是一致地使用UncaughtExceptionHandler来捕获；Native层的崩溃，也是一致地使用signal库来处理异常的信号量；对于ANR，Bugsnag和xCrash的处理方式一致，对于Android SDK 21以上都是通过处理SIGQUIT信号量来处理ANR，原来可参考ANR信号拦截与处理。

参考资料：Bugsnag-Android文档、美团Crash治理之路、得物App Crash治理演进、ANR信号拦截与处理、天猫启动安全模式