

---

# 高通用户体验性能优化期刊

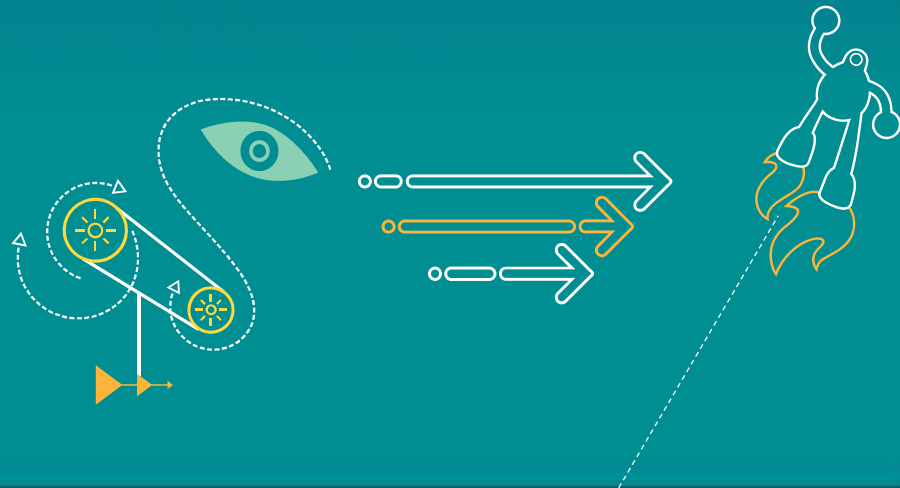
---



Qualcomm Technologies, Inc.

Confidential and Proprietary – Qualcomm Technologies, Inc.

机密和专有信息——高通技术股份有限公司



# Confidential and Proprietary – Qualcomm Technologies, Inc.

---

## Confidential and Proprietary – Qualcomm Technologies, Inc.

**NO PUBLIC DISCLOSURE PERMITTED:** Please report postings of this document on public servers or web sites to: [DocCtrlAgent@qualcomm.com](mailto:DocCtrlAgent@qualcomm.com). **禁止公开：**如在公共服务器或网站上发现本文档，请报告至：[DocCtrlAgent@qualcomm.com](mailto:DocCtrlAgent@qualcomm.com).

**Restricted Distribution:** Not to be distributed to anyone who is not an employee of either Qualcomm or its affiliated without the express approval of Qualcomm's Configuration Management. **限制分发：**未经高通配置管理部门的明示批准，不得发布给任何非高通或高通附属及关联公司员工的人。 Not to be used, copied, reproduced, or modified in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Technologies, Inc. 未经高通技术股份有限公司明示的书面允许，不得使用、复印、复制、或修改全部或部分文档，不得以任何形式向他人透露其内容。

The user of this documentation acknowledges and agrees that any Chinese text and/or translation herein shall be for reference purposes only and that in the event of any conflict between the English text and/or version and the Chinese text and/or version, the English text and/or version shall be controlling. 本文档的用户知悉并同意中文文本和/或翻译仅供参考之目的，如英文文本和/或版本和中文文本和/或版本之间存在冲突，以英文文本和/或版本为准。

This document contains confidential and proprietary information and must be shredded when discarded. 未经高通明示的书面允许，不得使用、复印、复制全部或部分文档，不得以任何形式向他人透露其内容。本文档含有高通机密和专有信息，丢弃时必须粉碎销毁。

Qualcomm reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed for any damages arising directly or indirectly by their use or application. The information provided in this document is provided on an "as is" basis. 高通保留未经通知即修改本档中提及的产品或信息的权利。本公司对使用或应用本文档所产生的直接或间接损失概不负责。本文档中的信息为基于现状所提供，使用风险由用户自行承担。

Qualcomm is a trademark of QUALCOMM Incorporated, registered in the United States and other countries. All QUALCOMM Incorporated trademarks are used with permission. Other product and brand names may be trademarks or registered trademarks of their respective owners. Qualcomm是高通公司在美国及其它国家注册的商标。所有高通公司的商标皆获得使用许可。其它产品和品牌名称可能为其各自所有者的商标或注册商标。

This technical data may be subject to U.S. and international export, re-export, or transfer ("export") laws. Diversion contrary to U.S. and international law is strictly prohibited. 本文档及所含技术资料可能受美国和国际出口、再出口或转移出口法律的 限制。严禁违反或偏离美国和国际的相关法律。

**Qualcomm Technologies, Inc. 5775 Morehouse Drive San Diego, CA 92121 U.S.A.**

高通技术股份有限公司，美国加利福尼亚州圣地亚哥市莫豪斯路 5775 号，邮编 92121

# Revision History

---

Revision	Date	Description
A	Mar 2016	Initial release

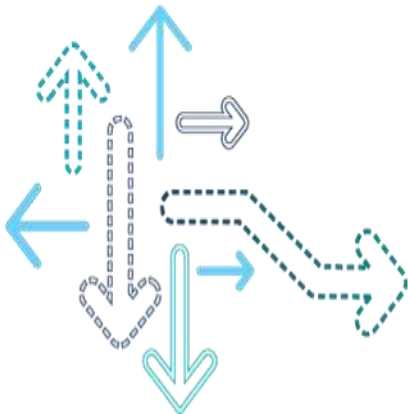
**Note:** There is no Rev. I, O, Q, S, X, or Z per Mil. standards.

# Contents

---

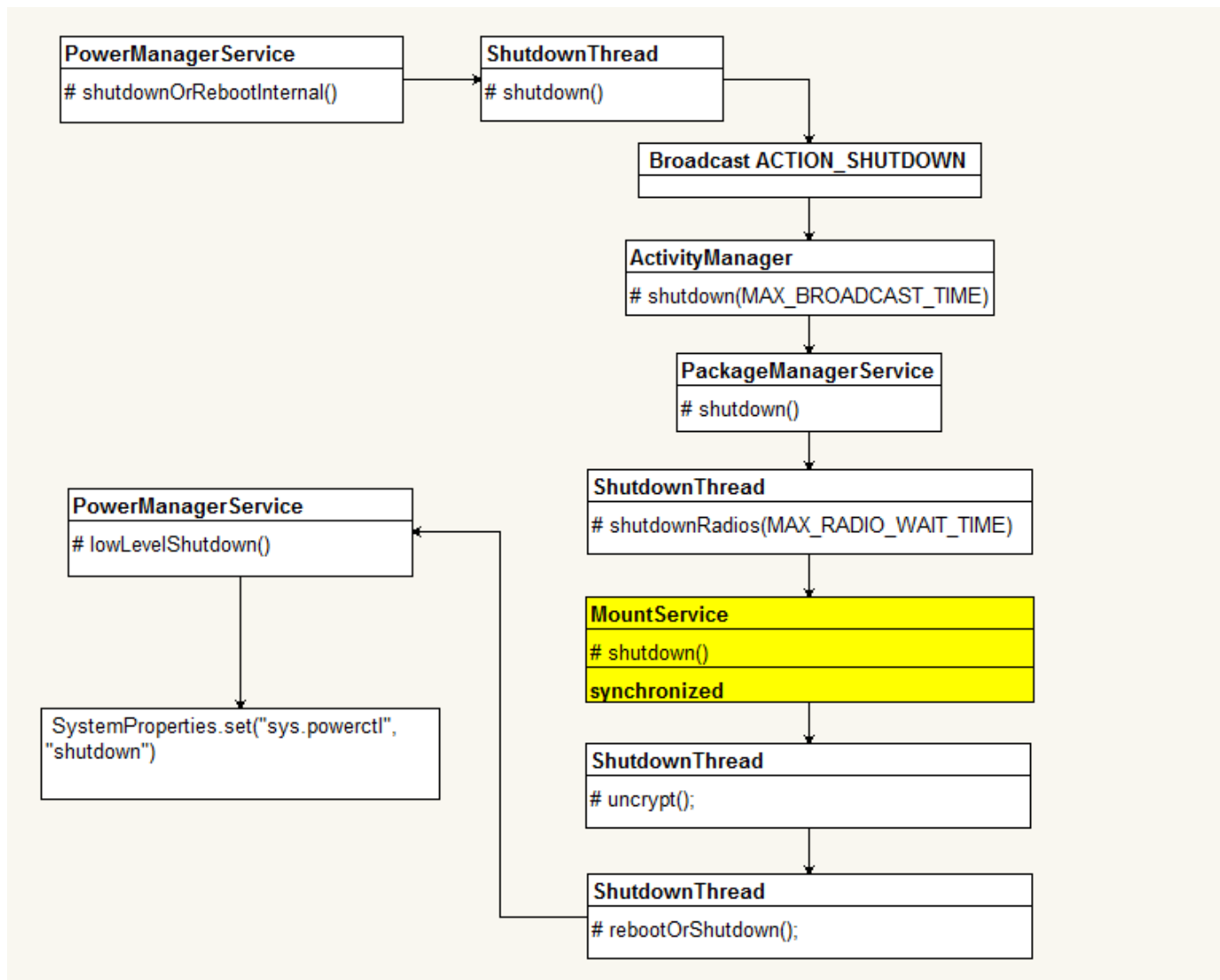
- Android关机慢问题分析
- Android设备唤醒时间分析
- ANDROID-25773204/CVE-2016-0805

## Android关机慢问题分析



# Android关机慢问题分析

## PowerManagerService关机流程



# Android关机慢问题分析

## 常见问题

### MountService 超时导致关机时间长

由于打开的应用较多，应用程序持有文件句柄，导致MountService在shutdownThread关机过程中超时。

```
25827: 01-10 02:33:01.992 1521 4255 I ShutdownThread: Shutting down MountService
34865: 01-10 02:33:21.993 1521 4255 W ShutdownThread: Shutdown wait timed out
```

可以使用ls -lsof 来查看有哪些应用打开了分区上的文件而没有释放

5529:	com.andro	2531	u0_a10	18	???	???	???	???	/storage/emulated/0/launcher.log
5549:	com.andro	2531	u0_a10	28	???	???	???	???	/storage/emulated/0/.sarrs_desktop/journal
5579:	com.andro	2531	u0_a10	43	???	???	???	???	/storage/emulated/0/.sarrs_desktop/journal
5937:	com.andro	2714	u0_a78	16	???	???	???	???	/storage/emulated/0/Android/data/com.android.music/cache/ui-images/journal
6853:	com.andro	3618	u0_a62	17	???	???	???	???	/storage/emulated/0/Android/data/com.letv.android.note/cache/ui-images/journal
7767:	com.andro	4083	u0_a108	16	???	???	???	???	/storage/emulated/0/superlive/.image/journal
7807:	com.andro	4105	u0_a1	17	???	???	???	???	/storage/emulated/0/feedback.log
7817:	com.andro	4105	u0_a1	23	???	???	???	???	/storage/emulated/0/recentLogs/2015_12_11-16_50_system_app_anr_log.zip
7819:	com.andro	4105	u0_a1	24	???	???	???	???	/storage/emulated/0/recentLogs/2015_12_11-16_50_system_app_anr_log.zip
7821:	com.andro	4105	u0_a1	25	???	???	???	???	/storage/emulated/0/recentLogs/2015_12_11-16_50_system_app_anr_log.zip
8071:	com.andro	4284	u0_a40	27	???	???	???	???	/storage/emulated/0/Android/data/com.android.deskclock/cache/ui-images/journal

# Android关机慢问题分析

---

## Solution1

应用监听“android.intent.action.REBOOT”和“android.intent.action.ACTION\_SHUTDOWN”，在收到此类消息时将所持文件句柄释放。

## Solution 2

针对第三方应用程序，可以调整ForceUnmount的sleep时间

代码路径 system/vold/Utils.cpp

**Note：** 由于此接口在doUmount()中并不检查返回值，因此并不存在返回失败的场景的处理。



# Android设备唤醒时间分析

# Android设备唤醒时间

---

拆分为两大部分

Kernel space的唤醒时间分析

从按下电源键(power)键，到中断qpnp\_kpdpwr\_status响应的时间

User space的唤醒时间分析

从中断qpnp\_kpdpwr\_status响应到第SurfaceFlinger第一帧绘制的时间

qpnp-power-on.c :  
qpnp\_kpdpwr\_irq

# Kernel space的唤醒时间分析

---

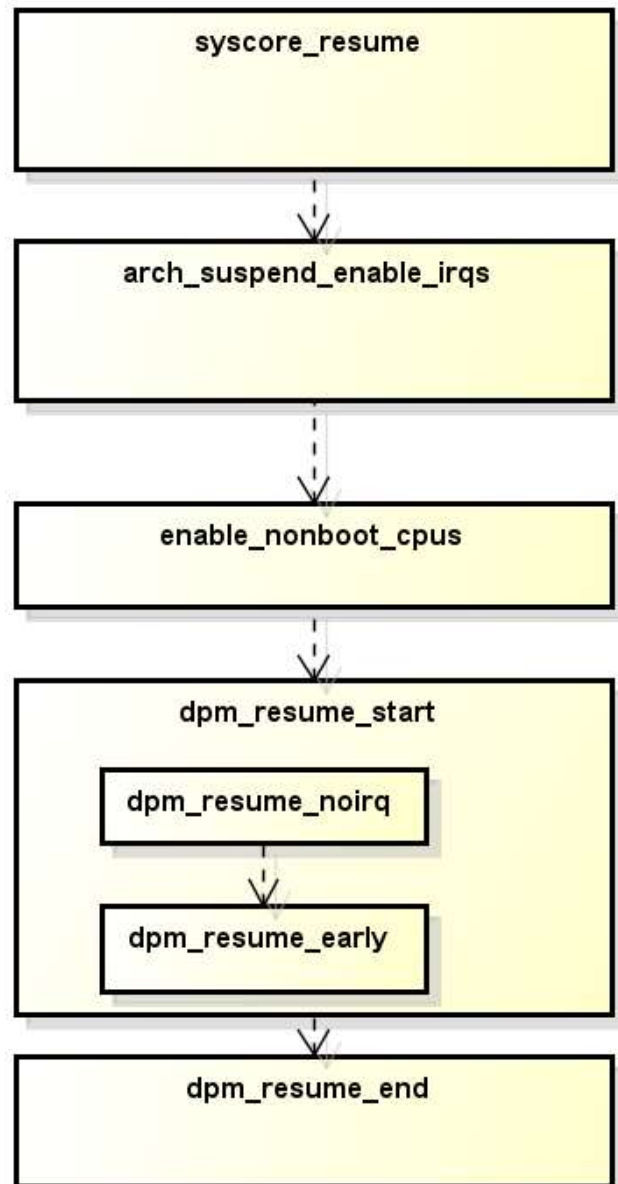
开启对应debug代码:

kernel config: CONFIG\_PM\_SLEEP\_DEBUG=y

device\qcom\msmxxx\BoardConfig.mk:

BOARD\_KERNEL\_CMDLINE += initcall\_debug log\_buf\_len=16M

# Kernel Space唤醒流程



- 对应的kernel log:
- dpm\_resume\_noirq完成  
[ 141.782368] PM: noirq resume of devices complete after 37.011 msecs
- dpm\_resume\_early完成  
[ 141.794448] PM: early resume of devices complete after 7.862 msecs
- dpm\_resume\_end  
[ 141.906757] PM: resume of devices complete after 112.295 msecs

# 时间分析

Kernel 中device的唤醒，打印如下格式：

call **device\_xxx** returned 0 after **time\_yyyy** usecs inf=**zzz\_type**

**device\_xxx**: 设备的名字

**time\_yyy**: 唤醒花费的时间，单位微妙

**zzz\_type**: 唤醒设备的类型及唤醒阶段，

如果某个唤醒阶段花费的时间较多，那么可以查找下表，分析对应的实现函数

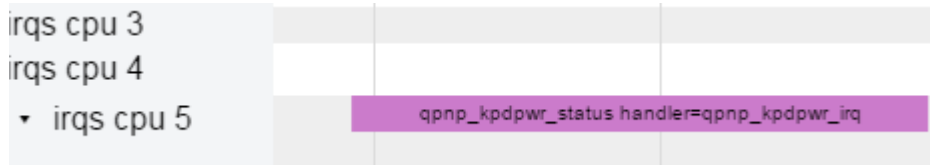
info字段	实现函数
noirq power domain	pm_domain->ops->resume_noirq
noirq class	class->pm->resume_noirq
noirq bus	bus->pm->resume_noirq
noirq driver	driver->pm->resume_noirq
early class	class->pm->resume_early
early bus	bus->pm->resume_early
early driver	driver->pm->resume_early
early class	class->resume
early bus	bus->pm->resume
early driver	driver->pm->resume

# User Space的唤醒时间分析

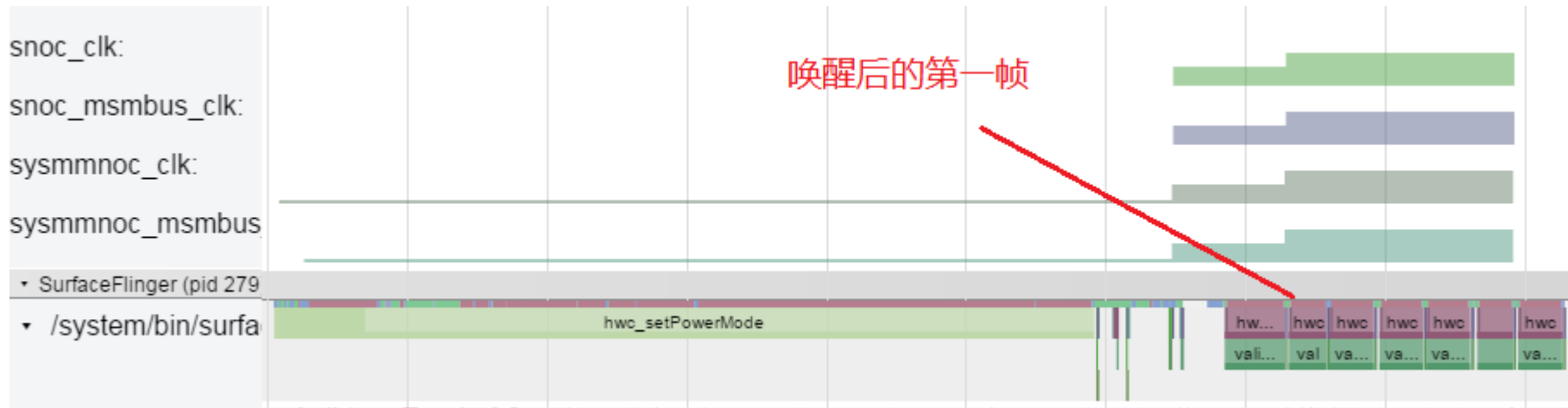
## 抓取Systrace与参考设备进行对比

- 1)连接usb线，按下power键，关闭屏幕，等1分钟
- 2)开始抓systrace，稍等2s后，按下power键，点亮屏幕

起始点：触发Power键中断，中断函数名字(`adb shell cat /proc/interrupts |grep kdpwr`)

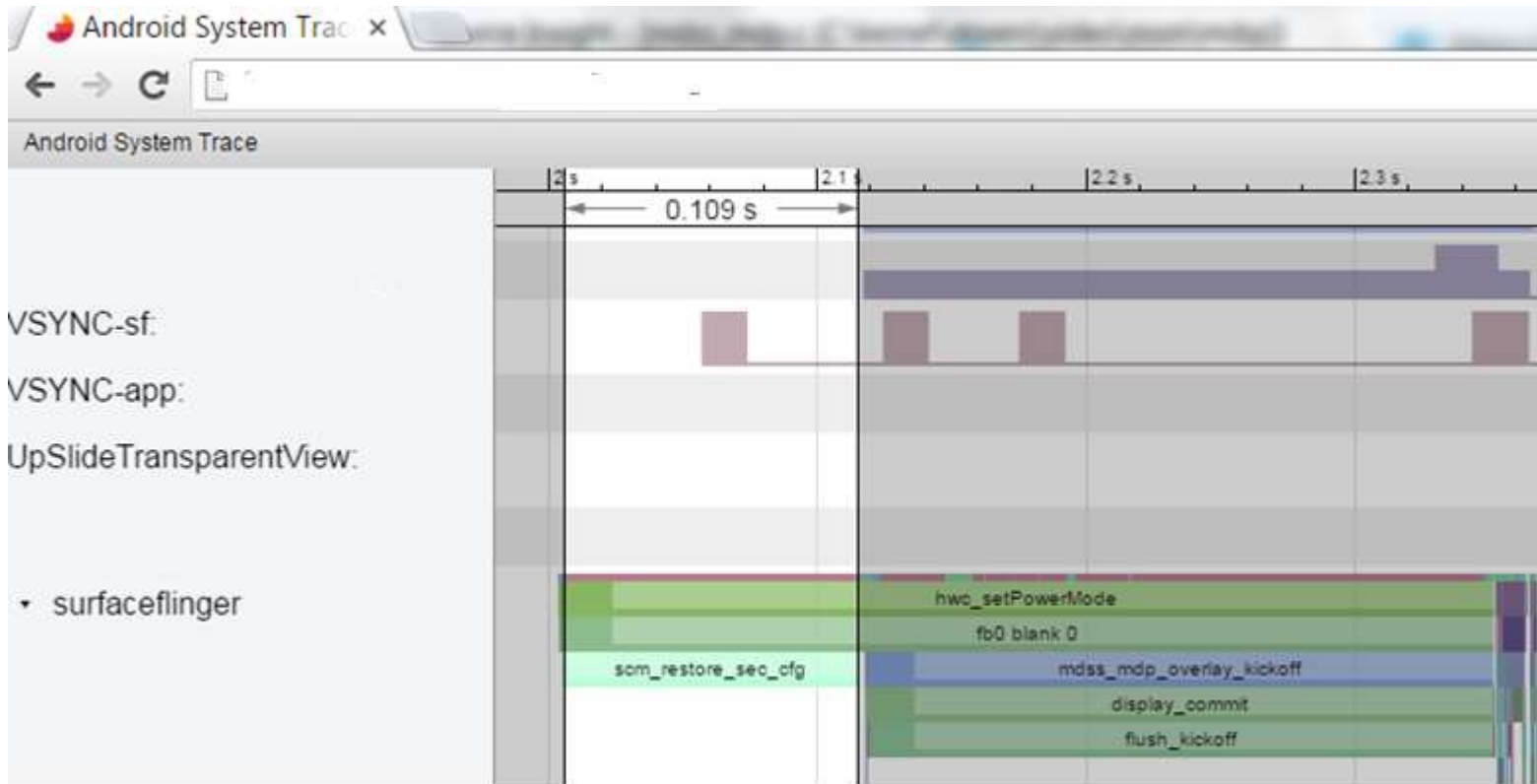


结束点：



# 典型案例:指纹唤醒慢

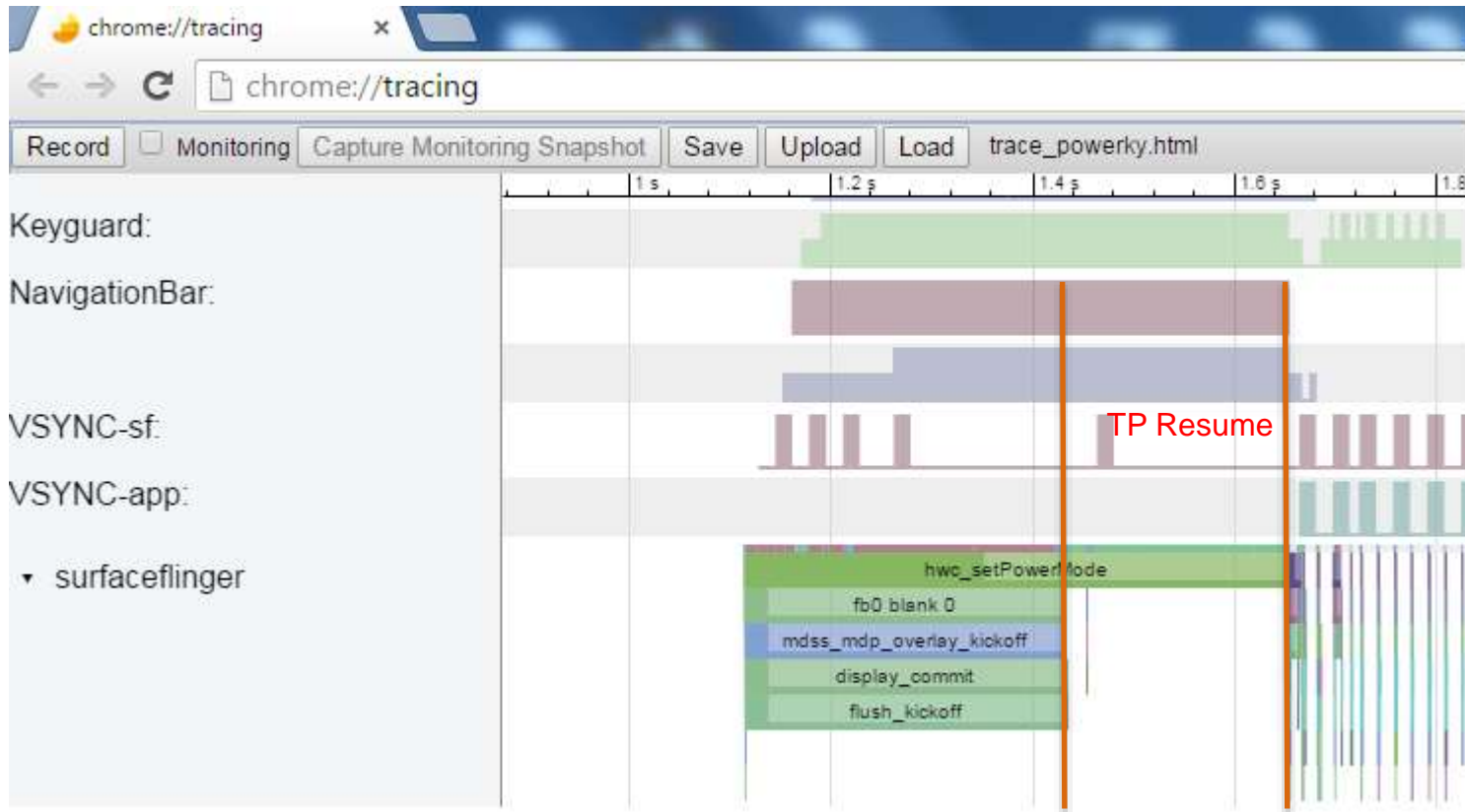
对比power键唤醒，发现指纹唤醒时间变长：  
进一步定位发现lcd唤醒与指纹唤醒会竞争信号量



解决方案：解决资源冲突，优先唤醒屏幕

# 典型案例:TP resume 慢

对比power键唤醒，发现TP resume 慢,典型的systrace log 如下.





## 典型案例:TP resume 慢

---

解决方案:

用优化TP driver 用workqueue 来实现TP 的resume. 请参考下面的实现.

<https://codeaurora.org/cgit/quic/la/kernel/msm-3.10/commit/?h=LA.BR.1.3.2&id=dd6e2392f921e3b3368a3e5d4f2885258a48d323>

<https://codeaurora.org/cgit/quic/la/kernel/msm-3.10/commit/?h=LA.BR.1.3.2&id=e291ac12f2e1ce2c0cb269b0b441d74c65a936c6>

**ANDROID-25773204/CVE-2016-0805**

# ANDROID-25773204/CVE-2016-0805

---

如果需要这个patch,请参考下面的link.

<https://www.codeaurora.org/cgit/quic/la/kernel/msm-3.10/commit/?h=msm-3.10&id=aca0f3abeca78c943a879b23c9445deac5fc9cd3>

---

## Questions?

<https://support.cdmatech.com>

