

Status:	CLOSED	Start date:	2023-05-25
Priority:	High	Due date:	2023-06-05
Assignee:	CD TEST-方永红	% Done:	100%
Category:	黑屏	Estimated time:	0.00 hour
Target version:			
Need_Info:	--	Found Version:	01.00.0100.c103
Resolution:	FIXED	Degrated:	--
Severity:	Critical	Verified Version:	
Reproducibility:	Occasionally	Fixed Version:	2023-05-26
Test Type:	ST	Root cause:	问题原因：问题的表现是黑屏和冻屏，但是引
Description			
环境信息】EE环境			
【monkey时间】：11小时			
【前提条件】：			
1.副屏上电			
2.台架环境完整			
3.monkey前网络时间声音均正常			
【操作步骤】			
1.执行monkey脚本			
【实际结果】			
1.副屏界面黑，有背光，显示触摸轨迹和顶部坐标，monkey已停止，触摸屏幕无任何反应，连接ADB投屏，可以操作界面返回（鼠标右键返回上一级界面，返回桌面），随后屏幕恢复，可正常点击，显示本次开机市场为11小时，monkey过程中未出现重启			
【期望结果】			
1.monkey正常运行，副屏无重启、死机现象			
【中控版本号】：03.02.0080.C101			
【副屏版本号】：01.00.0100.c103			
Subtasks:			
Bug # 118237: IOVDEV-29509】副屏】【0100】【台架】【monkey】monkey测试出现类似黑屏...			CLOSED
Bug # 118284: 【IOVDEV-29509】副屏】【0100】【台架】【monkey】monkey测试出现类似黑...			CLOSED

History

- #1 - 2023-05-22 11:37 - CDTS_TEST 王成
- http://cdiot@192.168.87.46/Pre_figure/Test_Log/Bug_118127/log
- #2 - 2023-05-22 14:39 - CD BSP-杜磊
- Status changed from New to ASSIGNED
- Assignee changed from CD BSP-杜磊 to 物联网项目组-RD3_CDTS 周飞

通过分析如下log，未发现屏幕和touch相关错误.

从问题现象和如下log分析，像是系统发生了卡顿/ANR, 过段时间后卡顿恢复正常。需要AF的同学帮忙再分析一下

```
tlog_sts40x190089_000090_0520082350.tar.gz
000090_manual_00001_230520_082245.log
05-20 06:18:20.561 25607 27676 E AndroidRuntime: FATAL EXCEPTION: ANR-Dump-Thread
05-20 06:18:20.561 25607 27676 E AndroidRuntime: Process: com.tencent.mm, PID: 25607
05-20 06:18:20.561 25607 27676 E AndroidRuntime: junit.framework.AssertionFailedError: Kernel not initialized by MMApplication!
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at junit.framework.Assert.fail(Assert.java:50)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at junit.framework.Assert.assertTrue(Assert.java:20)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at junit.framework.Assert.assertNotNull(Assert.java:218)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at m40.a0.h(Unknown Source:4)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at m40.a0.a(Unknown Source:8)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at yl.b.d(Unknown Source:29)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at yl.b$a$b.hasNext(Unknown Source:24)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.mm.sdk.event.EventCenter.queryStaticListenerByEvent(Unknown Source:34)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.mm.sdk.event.EventCenter.publish(Unknown Source:12)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.mm.sdk.event.IEvent.publish(Unknown Source:2)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.mm.app.o0.a(Unknown Source:659)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.matrix.trace.tracer.SignalAnrTracer.m(Unknown Source:22)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.matrix.trace.tracer.SignalAnrTracer.k(Unknown Source:128)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at com.tencent.matrix.trace.tracer.SignalAnrTracer$b.run(Unknown Source:149)
05-20 06:18:20.561 25607 27676 E AndroidRuntime:   at java.lang.Thread.run(Thread.java:920)
05-20 06:18:20.573 25905 25905 E AndroidRuntime: FATAL EXCEPTION: main
05-20 06:18:20.573 25905 25905 E AndroidRuntime: Process: com.bilibili.bilithings, PID: 25905
05-20 06:18:20.573 25905 25905 E AndroidRuntime: android.app.ForegroundServiceDidNotStartInTimeException: Context.startForegroundService() did not then call
Service.startForeground(): ServiceRecord{e3ee7a u0 com.bilibili.bilithings/com.bili.projection.service.ProjectionScreenService}
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.app.ActivityThread.throwRemoteServiceException(ActivityThread.java:1924)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.app.ActivityThread.access$2700(ActivityThread.java:248)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2149)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.os.Handler.dispatchMessage(Handler.java:106)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.os.Looper.loopOnce(Looper.java:201)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.os.Looper.loop(Looper.java:288)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at android.app.ActivityThread.main(ActivityThread.java:7880)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at java.lang.reflect.Method.invoke(Native Method)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:548)
05-20 06:18:20.573 25905 25905 E AndroidRuntime:   at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1009)
05-20 06:18:26.856 578 578 F libc : crash_dump helper failed to exec, or was killed
05-20 06:18:27.738 534 534 F libc : crash_dump helper failed to exec, or was killed
05-20 06:20:29.259 578 578 F libc : crash_dump helper failed to exec, or was killed
05-20 06:20:29.768 534 534 F libc : crash_dump helper failed to exec, or was killed
05-20 06:30:31.276 578 578 F libc : crash_dump helper failed to exec, or was killed
05-20 06:30:31.986 534 534 F libc : crash_dump helper failed to exec, or was killed
05-20 06:32:08.969 578 578 F libc : crash_dump helper failed to exec, or was killed
05-20 06:32:09.832 534 534 F libc : crash_dump helper failed to exec, or was killed
05-20 06:32:09.041 10454 10454 E AndroidRuntime: FATAL EXCEPTION: main
05-20 06:32:09.041 10454 10454 E AndroidRuntime: Process: com.android.music, PID: 10454
```

#3 - 2023-05-22 15:18 - CDTS_TEST 王成

- Assignee changed from 物联网项目组-RD3_CDTS 周飞到 CD FW 曹覃刚

#4 - 2023-05-22 16:22 - CD FW 曹覃刚

- Assignee changed from CD FW 曹覃刚 to CD FW 王武军

军哥，帮忙看一下这个问题

#5 - 2023-05-22 19:23 - CD FW 王武军

1、5月22日进展

- 1) 今天下午接手，刚开始分析，还在分析日志信息，从日志中可以看出系统发生了重启，详细的信息还需要进一步分析。
- 2、下一步
 - 1) 检查bsp同事提到的怀疑日志信息，分析那些日志是否是引起问题的原因。
 - 2) 从其他角度：watchdog、kernel等看是否会有所发现

#6 - 2023-05-22 20:33 - CD FW 曹覃刚

0520 早上八点19前后三分钟操作的屏幕

#7 - 2023-05-23 17:34 - CD FW 王武军

1、5月23日调查进展

(一) 首先bsp同事提到的ANR信息

所截取的ANR信息是05-20 06:18:20.561 - 05-20 06:32:09.041 是整个事件段的信息，与点击未响应的表现具体时间：
05-20 08:20:00差距很大，并不能说明当时的问题。
不过从ANR的角度来看，05-20 08:20:00的前10分钟确实存在很多的ANR问题：

Subject: executing service com.youku.phone/com.alibaba.analytics.AnalyticsService
----- Waiting Channels: pid 9956 at 2023-05-20 08:16:56.721697950+0800 -----
Cmd line: com.youku.phone:channel

Subject: Input dispatching timed out (Application does not have a focused window)
----- pid 11948 at 2023-05-20 08:17:12.329037319+0800 -----
Cmd line: com.tencent.qqlive

Subject: Input dispatching timed out (Application does not have a focused window)
----- pid 9225 at 2023-05-20 08:18:19.797483641+0800 -----
Cmd line: com.kugou.android.auto

Subject: Input dispatching timed out (Application does not have a focused window)
----- pid 11769 at 2023-05-20 08:18:30.214617543+0800 -----
Cmd line: com.huawei.appmarket

Subject: Context.startForegroundService() did not then call Service.startForeground(): ServiceRecord{907b1af u0
com.bilibili.bilithings/com.bili.projection.service.ProjectionScreenService}
----- pid 11220 at 2023-05-20 08:18:34.887607125+0800 -----
Cmd line: com.bilibili.bilithings

Subject: executing service com.ss.android.article.video.landscape/com.xiaomi.push.service.XMPushService
----- pid 10884 at 2023-05-20 08:18:35.121214885+0800 -----


```

05-20 08:19:08.345 6893 6893 I Monkey : // Injection Failed
05-20 08:19:08.346 6893 6893 I Monkey : Sleeping for 500 milliseconds
//1
05-20 08:19:08.465 1525 1731 E InputManager-JNI: An exception was thrown by callback 'notifyNoFocusedWindowAnr'.
05-20 08:19:08.465 1525 1731 E InputManager-JNI: java.lang.NullPointerException: Attempt to read from field 'java.lang.Object
com.android.server.wm.WindowProcessController.mOwner' on a null object reference
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at com.android.server.wm.ActivityRecord.inputDispatchingTimedOut(ActivityRecord.java:6282)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at com.android.server.wm.AnrController.notifyAppUnresponsive(AnrController.java:75)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at
com.android.server.wm.InputManagerCallback.notifyNoFocusedWindowAnr(InputManagerCallback.java:92)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at
com.android.server.input.InputManagerService.notifyNoFocusedWindowAnr(InputManagerService.java:2746)
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.469 29112 29112 I Monkey : // Injection Failed
05-20 08:19:08.470 29112 29112 I Monkey : :Sending Touch (ACTION_MOVE): 0:(751.8511,660.3587) 1:(240.51378,633.30096)
05-20 08:19:08.470 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.470 1525 20845 W InputManager: Input event injection from pid 15756 failed.
05-20 08:19:08.470 1525 17081 W InputManager: Input event injection from pid 8319 failed.
//2
05-20 08:19:08.471 1525 1627 W IPCThreadState: Waiting for thread to be free. mExecutingThreadsCount=32 mMaxThreads=31
05-20 08:19:08.471 8319 8319 I Monkey : // Injection Failed
05-20 08:19:08.471 8319 8319 I Monkey : :Sending Touch (ACTION_MOVE): 0:(1367.1829,958.7491)
05-20 08:19:08.471 15756 15756 I Monkey : // Injection Failed
05-20 08:19:08.472 1525 16914 W InputManager: Input event injection from pid 757 failed.
05-20 08:19:08.472 1525 20833 W InputManager: Input event injection from pid 32025 failed.
05-20 08:19:08.472 15756 15756 I Monkey : :Sending Touch (ACTION_MOVE): 0:(2367.8247,406.52283)
05-20 08:19:08.472 1525 20833 E IPCThreadState: binder thread pool (31 threads) starved for 5121 ms
05-20 08:19:08.472 1525 1807 W InputManager: Input event injection from pid 19072 failed.
05-20 08:19:08.472 1525 2619 W InputManager: Input event injection from pid 21401 failed.
05-20 08:19:08.472 757 757 I Monkey : // Injection Failed
//3
05-20 08:19:56.621 28656 28656 I Monkey : // Injection Failed
05-20 08:19:56.621 15475 15475 I Monkey : // Injection Failed
05-20 08:19:56.621 28656 28656 I Monkey : :Sending Touch (ACTION_POINTER_DOWN 1): 0:(2060.9102,131.76442) 1:(377.0,90.0)
05-20 08:19:56.621 15475 15475 I Monkey : :Sending Touch (ACTION_UP): 0:(1436.6306,380.53757)
05-20 08:19:56.623 8436 8436 W Monkey : ** Error: A RuntimeException occurred:
05-20 08:19:56.624 15642 15642 W Monkey : ** Error: A RuntimeException occurred:
05-20 08:19:56.624 8436 8436 W Monkey : java.lang.SecurityException: Injecting to another application requires INJECT_EVENTS permission
05-20 08:19:56.624 8436 8436 W Monkey : at android.os.Parcel.createExceptionOrNull(Parcel.java:2425)
05-20 08:19:56.624 8436 8436 W Monkey : at android.os.Parcel.createException(Parcel.java:2409)
05-20 08:19:56.624 8436 8436 W Monkey : at android.os.Parcel.readException(Parcel.java:2392)
05-20 08:19:56.624 8436 8436 W Monkey : at android.os.Parcel.readException(Parcel.java:2334)
05-20 08:19:56.624 8436 8436 W Monkey : at android.hardware.input.IInputManager$Stub$Proxy.injectInputEvent(IInputManager.java:1379)
05-20 08:19:56.624 8436 8436 W Monkey : at android.hardware.input.InputManager.injectInputEvent(InputManager.java:1091)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.commands.monkey.MonkeyMotionEvent.injectEvent(MonkeyMotionEvent.java:188)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.commands.monkey.Monkey.runMonkeyCycles(Monkey.java:1217)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.commands.monkey.Monkey.run(Monkey.java:708)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.commands.monkey.Monkey.main(Monkey.java:568)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.internal.os.RuntimeInit.nativeFinishInit(Native Method)
05-20 08:19:56.624 8436 8436 W Monkey : at com.android.internal.os.RuntimeInit.main(RuntimeInit.java:355)
05-20 08:19:56.625 15642 15642 W Monkey : java.lang.SecurityException: Injecting to another application requires INJECT_EVENTS permission

```

```

05-20 08:19:56.625 15642 15642 W Monkey : at android.os.Parcel.createExceptionOrNull(Parcel.java:2425)
05-20 08:19:56.625 15642 15642 W Monkey : at android.os.Parcel.createException(Parcel.java:2409)
05-20 08:19:56.625 15642 15642 W Monkey : at android.os.Parcel.readException(Parcel.java:2392)
05-20 08:19:56.625 15642 15642 W Monkey : at android.os.Parcel.readException(Parcel.java:2334)
05-20 08:19:56.625 15642 15642 W Monkey : at android.hardware.input.IInputManager$Stub$Proxy.injectInputEvent(IInputManager.java:1379)
05-20 08:19:56.625 15642 15642 W Monkey : at android.hardware.input.InputManager.injectInputEvent(InputManager.java:1091)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.commands.monkey.MonkeyMotionEvent.injectEvent(MonkeyMotionEvent.java:188)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.commands.monkey.Monkey.runMonkeyCycles(Monkey.java:1217)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.commands.monkey.Monkey.run(Monkey.java:708)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.commands.monkey.Monkey.main(Monkey.java:568)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.internal.os.RuntimeInit.nativeFinishInit(Native Method)
05-20 08:19:56.625 15642 15642 W Monkey : at com.android.internal.os.RuntimeInit.main(RuntimeInit.java:355)
05-20 08:19:56.625 8436 8436 I Monkey : Events injected: 64083
05-20 08:19:56.625 15642 15642 I Monkey : Events injected: 38297
05-20 08:19:56.626 27494 27494 W Monkey : ** Error: A RuntimeException occurred:
05-20 08:19:56.626 15642 15642 I Monkey : :Sending rotation degree=0, persist=false
05-20 08:19:56.626 8436 8436 I Monkey : :Sending rotation degree=0, persist=false
05-20 08:19:56.626 2725 2725 W Monkey : ** Error: A RuntimeException occurred:
05-20 08:19:56.626 27494 27494 W Monkey : java.lang.SecurityException: Injecting to another application requires INJECT_EVENTS permission
05-20 08:19:56.626 27494 27494 W Monkey : at android.os.Parcel.createExceptionOrNull(Parcel.java:2425)
05-20 08:19:56.626 27494 27494 W Monkey : at android.os.Parcel.createException(Parcel.java:2409)
05-20 08:19:56.626 27494 27494 W Monkey : at android.os.Parcel.readException(Parcel.java:2392)
05-20 08:19:56.626 27494 27494 W Monkey : at android.os.Parcel.readException(Parcel.java:2334)
05-20 08:19:56.626 27494 27494 W Monkey : at android.hardware.input.IInputManager$Stub$Proxy.injectInputEvent(IInputManager.java:1379)
05-20 08:19:56.626 27494 27494 W Monkey : at android.hardware.input.InputManager.injectInputEvent(InputManager.java:1091)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.commands.monkey.MonkeyMotionEvent.injectEvent(MonkeyMotionEvent.java:188)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.commands.monkey.Monkey.runMonkeyCycles(Monkey.java:1217)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.commands.monkey.Monkey.run(Monkey.java:708)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.commands.monkey.Monkey.main(Monkey.java:568)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.internal.os.RuntimeInit.nativeFinishInit(Native Method)
05-20 08:19:56.626 27494 27494 W Monkey : at com.android.internal.os.RuntimeInit.main(RuntimeInit.java:355)
05-20 08:19:56.627 27494 27494 I Monkey : Events injected: 68592

```

在整个持续的失败过程中，发现有3个异常点：

1) 在monkey失败的过程中，有出现InputManager-JNI的异常日志

```

05-20 08:19:08.465 1525 1731 E InputManager-JNI: An exception was thrown by callback 'notifyNoFocusedWindowAnr'.
05-20 08:19:08.465 1525 1731 E InputManager-JNI: java.lang.NullPointerException: Attempt to read from field 'java.lang.Object
com.android.server.wm.WindowProcessController.mOwner' on a null object reference
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at com.android.server.wm.ActivityRecord.inputDispatchingTimedOut(ActivityRecord.java:6282)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at com.android.server.wm.AnrController.notifyAppUnresponsive(AnrController.java:75)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at
com.android.server.wm.InputManagerCallback.notifyNoFocusedWindowAnr(InputManagerCallback.java:92)
05-20 08:19:08.465 1525 1731 E InputManager-JNI: at
com.android.server.input.InputManagerService.notifyNoFocusedWindowAnr(InputManagerService.java:2746)
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.
05-20 08:19:08.466 1525 1731 I InputDispatcher: Dropped event because it is stale.

```

大致可以看出是，InputManager在响应底层ANR的分发时出现了回调的空指针，应该与InputDispatcher的anr事件回调有关系。

2) 05-20 08:19:08.471 1525 1627 W IPCThreadState: Waiting for thread to be free. mExecutingThreadsCount=32 mMaxThreads=31
IPC的线程可执行数超出了，最大执行数，那么就意味着后续的IPC线程可能就是被挂起的并未及时的执行相关的逻辑

3) 05-20 08:19:56.626 27494 27494 W Monkey : java.lang.SecurityException: Injecting to another application requires INJECT_EVENTS permission
可以看出是monkey在事件注入时缺少了INJECT_EVENTS的权限，导致monkey的注入出现了异常

因为在黑屏或者屏幕没有响应的整个时间段，，monkey一直在注入事件，但是事件并未被正常响应，所以屏幕并未变化或者出现黑屏，期间即使主动点击，事件也是不会被响应的，整个期间

事件分发应该都是被monkey的事件阻塞了，手动点击操作，就不会有响应，日志表现：

05-20 08:20:01.705 1525 1731 I InputDispatcher: Dropping event because there is no touched foreground window in display 0 or gesture monitor to receive it.

(2) monkey事件注入成功

从05-20 08:20:02.531这个时间开始，monkey的响应又回复了正常

05-20 08:20:02.531 11285 11356 I Monkey : // Allowing start of Intent { act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=com.android.music/.VideoBrowserActivity } in package com.android.music

05-20 08:20:02.536 1525 24843 I wm_task_moved: [3057,1,37]

05-20 08:20:02.536 1525 24843 I wm_task_to_front: [0,3057]

05-20 08:20:02.537 1525 24843 I wm_focused_root_task: [0,0,3057,3064,bringingFoundTaskToFront]

05-20 08:20:02.538 1525 24843 I wm_set_resumed_activity: [0,com.android.music/.VideoBrowserActivity,bringingFoundTaskToFront]

目前的分析小结

该票中提到的问题：重启、黑屏、冻屏，从日志中都大致找到了对应的点，但是现在还没有办法完全定位出

这一些列的问题发生的根本原因。目前的怀疑点是，三方应用在monkey的测试过程中因为某一些异常出现不断重启，导致系统异常重启，进而响应了monkey本身的事件注入，而monkey事件的注入引起了Input分发系统的异常，导致了屏幕点击没响应和黑屏的可能。

2、下一步

1) 测试方面：是否可以做一个对比测试：去掉目前引起系统重启的三方应用，继续跑monkey看是否会出现类似的问题；

2) 研发方面：继续从日志入手，强化分析

#8 - 2023-05-23 19:06 - CDTs_TEST 王成

- Category set to BSP

#9 - 2023-05-23 21:16 - CDTs_TEST 王成

- Due date set to 2023-05-26

#10 - 2023-05-23 21:34 - CDTs_TEST 王成

- Fixed Version set to 2023-05-26

#11 - 2023-05-24 10:16 - CDTs-TEST 周婷

- Category changed from BSP to 黑屏

#12 - 2023-05-24 17:04 - CD FW王武军

1、5月24日进展

1) 通过昨天针对现象和日志信息的初步分析，整个过程涉及到了ANR问题、冻屏问题，

从问题的发生的先后顺序来看，ANR的发生先于冻屏问题，所以今天的调查方向是从ANR问题入手。

在发生冻屏问题（05-20 08:19:00）的前后，有很多ANR发生，从分析来看：

在system_server发生ANR（anr_2023-05-20-08-12-40-511）的前后

都有很多应用发生了ANR，类型不尽相同，但是原因大致都差不多，都是在进行IPC通信时线程请求被阻塞了。

具体信息如下：

system_server发生ANR之前：

ANR文件：anr_2023-05-20-08-10-03-099

对应的应用：com.youku.phone

ANR文件：anr_2023-05-20-08-10-23-651

对应的应用：com.tencent.mm

ANR文件：anr_2023-05-20-08-10-54-673

对应的应用：com.android.music

ANR文件：anr_2023-05-20-08-11-09-565

对应的应用：com.bilibili.bilithings

ANR文件：anr_2023-05-20-08-11-18-236

对应的应用：com.android.settings

ANR文件：anr_2023-05-20-08-11-24-893

对应的应用：com.youku.phone:channel

都存在类似的dump信息，IPCThreadState通信被阻塞

native: #00 pc 000715a8 /apex/com.android.runtime/lib/bionic/libc.so (__ioctl+8)

native: #01 pc 0003f2c7 /apex/com.android.runtime/lib/bionic/libc.so (ioctl+26)

native: #02 pc 0003abff /system/lib/libbinder.so (android::IPCThreadState::talkWithDriver(bool)+226)

native: #03 pc 0003b9f1 /system/lib/libbinder.so (android::IPCThreadState::waitForResponse(android::Parcel*, int*)+124)

native: #04 pc 0003b747 /system/lib/libbinder.so (android::IPCThreadState::transact(int, unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+134)

native: #05 pc 00035f3b /system/lib/libbinder.so (android::BpBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+118)

native: #06 pc 000f0201 /system/lib/libandroid_runtime.so (android_os_BinderProxy_transact(_JNIEnv*, _jobject*, int, _jobject*, _jobject*, int)+84)

at android.os.BinderProxy.transactNative(Native method)

at android.os.BinderProxy.transact(BinderProxy.java:571)

system_server发生ANR之后：

ANR文件：anr_2023-05-20-08-14-24-195

对应的应用：com.kugou.framework.service

ANR文件：anr_2023-05-20-08-15-38-749

对应的应用：com.thundercomm.gamecenter

ANR文件：anr_2023-05-20-08-15-43-875

对应的应用：com.sohu.inputmethod.sogou.car

ANR文件：anr_2023-05-20-08-16-24-969

对应的应用：com.youku.phone

ANR文件：anr_2023-05-20-08-16-56-721

对应的应用：com.youku.phone:channel

ANR文件：anr_2023-05-20-08-17-12-325

对应的应用：com.tencent.qqlive

ANR文件：anr_2023-05-20-08-18-19-793

对应的应用：com.kugou.android.auto

ANR文件：anr_2023-05-20-08-18-30-210

对应的应用：com.huawei.appmarket

ANR文件：anr_2023-05-20-08-18-34-879

对应的应用：com.bilibili.bilithings

都存在类似的dump信息，IPCThreadState通信被阻塞

native: #02 pc 000000000053070 /system/lib64/libbinder.so (android::IPCThreadState::talkWithDriver(bool)+292)

native: #03 pc 0000000000542ac /system/lib64/libbinder.so (android::IPCThreadState::waitForResponse(android::Parcel*, int*)+64)

native: #04 pc 000000000053ff0 /system/lib64/libbinder.so (android::IPCThreadState::transact(int, unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+224)

native: #05 pc 00000000004bafe /system/lib64/libbinder.so (android::BpBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+196)

native: #06 pc 0000000000167664 /system/lib64/libandroid_runtime.so (android_os_BinderProxy_transact(_JNIEnv*, _jobject*, int, _jobject*, _jobject*, int)+156)

at android.os.BinderProxy.transactNative(Native method)

at android.os.BinderProxy.transact(BinderProxy.java:571)

2) 分析system_server的ANR信息

ANR文件：anr_2023-05-20-08-12-40-511

对应的应用：system_server

"main" prio=5 tid=1 Native

native: #00 pc 0000000000a2c4c /apex/com.android.runtime/lib64/bionic/libc.so (__epoll_pwait+12)

native: #01 pc 000000000017ea8 /system/lib64/libutils.so (android::Looper::pollInner(int)+184)

native: #02 pc 000000000017d84 /system/lib64/libutils.so (android::Looper::pollOnce(int, int*, int*, void**)+116)

native: #03 pc 0000000000152d3c /system/lib64/libandroid_runtime.so (android::android_os_MessageQueue_nativePollOnce(_JNIEnv*, _jobject*, long, int)+48)

at android.os.MessageQueue.nativePollOnce(Native method)

at android.os.MessageQueue.next(MessageQueue.java:335)

at android.os.Looper.loopOnce(Looper.java:161)

at android.os.Looper.loop(Looper.java:288)

"android.ui" prio=5 tid=12 Blocked

at com.android.server.wm.ActivityTaskManagerService.getTasks(ActivityTaskManagerService.java:2250)

- waiting to lock <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)

at com.android.server.wm.DisplayPolicy.getAppPackageName(DisplayPolicy.java:443)

"android.display" prio=5 tid=14 Blocked

at com.android.server.wm.WindowManagerService\$H.handleMessage(WindowManagerService.java:5326)

- waiting to lock <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)

at android.os.Handler.dispatchMessage(Handler.java:106)

"Binder:1525_20" prio=5 tid=160 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x13c14b58 self=0xb40000757407b9f0

| sysTid=26556 nice=-10 cgrp=default sched=0/0 handle=0x72d7086cb0

| state=S schedstat=(306246980651 365496457548 2192152) utm=20619 stm=10005 core=3 HZ=100

| stack=0x72d6f8f000-0x72d6f91000 stackSize=991KB

| held mutexes=

native: #00 pc 0000000000a1c88 /apex/com.android.runtime/lib64/bionic/libc.so (__ioctl+8)

native: #01 pc 000000000005b8d4 /apex/com.android.runtime/lib64/bionic/libc.so (ioctl+156)

native: #02 pc 0000000000053070 /system/lib64/libbinder.so (android::IPCThreadState::talkWithDriver(bool)+292)

native: #03 pc 00000000000542ac /system/lib64/libbinder.so (android::IPCThreadState::waitForResponse(android::Parcel*, int*)+64)

native: #04 pc 0000000000053ff0 /system/lib64/libbinder.so (android::IPCThreadState::transact(int, unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+224)

native: #05 pc 000000000004bafc /system/lib64/libbinder.so (android::BpBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+196)

native: #06 pc 0000000000167664 /system/lib64/libandroid_runtime.so (android_os_BinderProxy_transact(_JNIEnv*, _jobject*, int, _jobject*, _jobject*, int)+156)

at android.os.BinderProxy.transactNative(Native method)

at android.os.BinderProxy.transact(BinderProxy.java:571)

at android.app.IActivityController\$Stub\$Proxy.activityStarting(IActivityController.java:275)

at com.android.server.wm.ActivityStarter.executeRequest(ActivityStarter.java:1017)

at com.android.server.wm.ActivityStarter.execute(ActivityStarter.java:671)

- locked <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)

at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1227)

ANR文件：anr_2023-05-20-08-20-06-979

对应的应用：system_server

at com.android.server.wm.WindowAnimator.lambda\$new\$1\$WindowAnimator(WindowAnimator.java:103)

```

- waiting to lock <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)
at com.android.server.wm.WindowAnimator$$ExternalSyntheticLambda0.doFrame(unavailable:-1)
at com.android.server.wm.ActivityStarter.execute(ActivityStarter.java:671)
- locked <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)
at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1227)
ANR文件：anr_2023-05-20-08-20-30-494
    对应的应用：system_server
at com.android.server.thunderperf.ThunderBoostPerfService$1.updateResolution(ThunderBoostPerfService.java:109)
- waiting to lock <0x06e4d00f> (a java.lang.Object)
at com.thundercomm.perf.BoostPerfManager.updateResolution(BoostPerfManager.java:71)

at libcore.io.Linux.open(Native method)
at libcore.io.ForwardingOs.open(ForwardingOs.java:567)
at libcore.io.BlockGuardOs.open(BlockGuardOs.java:273)
at libcore.io.IoBridge.open(IoBridge.java:561)
at java.io.FileOutputStream.<init>(FileOutputStream.java:236)
at java.io.FileOutputStream.<init>(FileOutputStream.java:186)
at java.io.PrintWriter.<init>(PrintWriter.java:263)
at com.android.server.compat.CompatConfig.saveOverrides(CompatConfig.java:741)
- locked <0x084666f7> (a java.io.File)
at com.android.server.compat.CompatConfig.addOverrides(CompatConfig.java:263)
at com.android.server.compat.PlatformCompat.setOverrides(PlatformCompat.java:225)
at com.android.server.thunderperf.ThunderBoostPerfService.checkAndSetResolution(ThunderBoostPerfService.java:183)
at com.android.server.thunderperf.ThunderBoostPerfService.setResolution(ThunderBoostPerfService.java:117)
at com.android.server.thunderperf.ThunderBoostPerfService.access$300(ThunderBoostPerfService.java:52)
at com.android.server.thunderperf.ThunderBoostPerfService$1.updateResolution(ThunderBoostPerfService.java:110)
- locked <0x06e4d00f> (a java.lang.Object)
at com.thundercomm.perf.BoostPerfManager.updateResolution(BoostPerfManager.java:71)
at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1190)

```

可以发现system_server进程中存在 0x05476839 对象互锁的情况，这样就会导致正常的逻辑不能被执行还有就是binder进程就不能被正常释放，后续其他进行

针对binder的使用就有被挂起的可能，分析0x05476839这个锁的持有情况来看是：ActivityTaskManagerService.startActivityAsUser 方法有关系。

针对0x06e4d00f也存在互锁的情况，有关的方法是：BoostPerfManager.updateResolution相关的逻辑。

综合小结：ANR的发生与system_server进程中存在互锁调用有关系，导致binder被阻塞，其他应用需要申请binder请求时被挂起，导致anr的发生。

2、下一步

- 1) 针对已经发生的互锁问题，调查解决方案，可能需要通过使用全局的锁来处理，具体的方案还需要分析。
- 2) 互锁问题处理后，可以避免类似的anr发生，然后check票中提到的冻屏是否还会发生。

先继续定位原因，然后测试协助装27个APK 尝试复现

#14 - 2023-05-25 13:42 - CD FW 曹覃刚

- Assignee changed from CD FW 王武军 to CD FW 郑雷

#15 - 2023-05-25 16:57 - CD FW 郑雷

[当前状态]

分析tlog_sts40x190089_000090_0520082350日志。

1.bugreport中一直有youku和audioserver一直在不停的出现crash。

系统会dump crash的信息，会造成cpu和io占用比较高。

关于youku的崩溃是youku app的问题。audioserver的崩溃会造成系统的重启。所以audioserver的崩溃需要audio的team看一下。

05-20 04:58:53.204 10051 19804 19804 E AndroidRuntime: FATAL EXCEPTION: main

05-20 04:58:53.204 10051 19804 19804 E AndroidRuntime: Process: com.thundercomm.screensaver, PID: 19804

05-20 04:58:53.204 10051 19804 19804 E AndroidRuntime: DeadSystemException: The system died; earlier logs will point to the root cause

05-20 05:03:13.830 1041 30829 30829 F DEBUG : *

05-20 05:03:13.830 1041 30829 30829 F DEBUG : Build fingerprint: 'HiPhi/PassengerDisplay_CN/PassengerDisplay:12/SKQ1.220201.001/3388:user/test-keys'

05-20 05:03:13.830 1041 30829 30829 F DEBUG : Revision: '0'

05-20 05:03:13.830 1041 30829 30829 F DEBUG : ABI: 'arm'

05-20 05:03:13.830 1041 30829 30829 F DEBUG : Timestamp: 2023-05-20 05:03:13.448272214+0800

05-20 05:03:13.830 1041 30829 30829 F DEBUG : Process uptime: 0s

05-20 05:03:13.830 1041 30829 30829 F DEBUG : Cmdline: /vendor/bin/hw/android.hardware.audio.service

05-20 05:03:13.830 1041 30829 30829 F DEBUG : pid: 29335, tid: 29335, name: audio.service >>> /vendor/bin/hw/android.hardware.audio.service <<<

05-20 05:03:13.830 1041 30829 30829 F DEBUG : uid: 1041

05-20 05:03:13.830 1041 30829 30829 F DEBUG : signal 35 (<debuggerd signal>), code 1 (SI_QUEUE from pid 29336, uid 1041), fault addr -----

05-20 05:03:13.830 1041 30829 30829 F DEBUG : r0 00000004 r1 c0306201 r2 fff89808 r3 fff8975c

05-20 05:03:13.830 1041 30829 30829 F DEBUG : r4 0f572bb6 r5 f0f1a138 r6 fff89808 r7 00000036

05-20 05:03:13.830 1041 30829 30829 F DEBUG : r8 00000000 r9 f0782c44 r10 fff89818 r11 fff89808

05-20 05:03:13.830 1041 30829 30829 F DEBUG : ip 00000100 sp fff89740 lr f0ec82cb pc f0efa5a8

05-20 05:03:13.830 1041 30829 30829 F DEBUG : backtrace:

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #00 pc 000715a8 /apex/com.android.runtime/lib/bionic/libc.so (_ioctl+8) (BuildId: cc194ef76ee932851b23cc2e4f46d55f)

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #01 pc 0003f2c7 /apex/com.android.runtime/lib/bionic/libc.so (ioctl+26) (BuildId: cc194ef76ee932851b23cc2e4f46d55f)

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #02 pc 0005ff09 /apex/com.android.vndk.v30/lib/libhidlbase.so

(android::hardware::IPCThreadState::getAndExecuteCommand()+160) (BuildId: ae84f31d0a55afdbe06ed13729dae8a2)

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #03 pc 0006100d /apex/com.android.vndk.v30/lib/libhidlbase.so

(android::hardware::IPCThreadState::joinThreadPool(bool)+56) (BuildId: ae84f31d0a55afdbe06ed13729dae8a2)

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #04 pc 000026b5 /vendor/bin/hw/android.hardware.audio.service (main+2564) (BuildId: b5ba2845ffd2ad1c80c85239344b8c71)

05-20 05:03:13.830 1041 30829 30829 F DEBUG : #05 pc 0003248b /apex/com.android.runtime/lib/bionic/libc.so (_libc_init+54) (BuildId: cc194ef76ee932851b23cc2e4f46d55f)

05-20 05:03:14.098 1041 29336 29373 F libc : Fatal signal 6 (SIGABRT), code 1 (SI_QUEUE) in tid 29373 (TimeCheckThread), pid 29336 (audioserver)

05-20 05:03:15.694 1041 30856 30856 F DEBUG : -----

05-20 05:03:15.694 1041 30856 30856 F DEBUG : Build fingerprint: 'HiPhi/PassengerDisplay_CN/PassengerDisplay:12/SKQ1.220201.001/3388:user/test-keys'

05-20 05:03:15.694 1041 30856 30856 F DEBUG : Revision: '0'

05-20 05:03:15.694 1041 30856 30856 F DEBUG : ABI: 'arm64'

05-20 05:03:15.694 1041 30856 30856 F DEBUG : Timestamp: 2023-05-20 05:03:14.211933203+0800

05-20 05:03:15.694 1041 30856 30856 F DEBUG : Process uptime: 0s

05-20 05:03:15.694 1041 30856 30856 F DEBUG : Cmdline: /system/bin/audioserver

```

05-20 05:03:15.694 1041 30856 30856 F DEBUG : pid: 29336, tid: 29373, name: TimeCheckThread >>> /system/bin/audioserver <<<
05-20 05:03:15.694 1041 30856 30856 F DEBUG : uid: 1041
05-20 05:03:15.694 1041 30856 30856 F DEBUG : signal 6 (SIGABRT), code -1 (SI_QUEUE), fault addr -----
05-20 05:03:15.694 1041 30856 30856 F DEBUG : Abort message: 'TimeCheck timeout for IAudioFlinger command 1'
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x0 0000000000000000 x1 00000000000072bd x2 0000000000000006 x3 0000007bc030f530
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x4 6d686b456e686374 x5 6d686b456e686374 x6 6d686b456e686374 x7 7f7f7f7f7f7f7f7f
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x8 00000000000000f0 x9 e234de4b90e2afb1 x10 0000000000000000 x11 fffff80fffffd
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x12 0000000000000001 x13 000000000000002e x14 0000007bc030f6c0 x15 0000000034155555
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x16 0000007ee7ed9060 x17 0000007ee7eb5560 x18 0000007bc00ae000 x19 0000000000007298
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x20 00000000000072bd x21 00000000ffffff x22 b400007bfdcffb50 x23 b400007bddd09714
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x24 b400007bddd09714 x25 0000007bc030fcb0 x26 0000007bc030fff8 x27 000000000000fc000
05-20 05:03:15.694 1041 30856 30856 F DEBUG : x28 0000007bc0217000 x29 0000007bc030f5b0
05-20 05:03:15.694 1041 30856 30856 F DEBUG : lr 0000007ee7e6595c sp 0000007bc030f510 pc 0000007ee7e65988 pst 0000000000001000
05-20 05:03:15.694 1041 30856 30856 F DEBUG : backtrace:
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #00 pc 0000000000051988 /apex/com.android.runtime/lib64/bionic/libc.so (abort+168) (BuildId:
369edc656806aeaf384cbeb8f7a347af)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #01 pc 00000000000063b8 /system/lib64/liblog.so (_android_log_default_aborter+16) (BuildId:
467c2038cdfa767245f9280e657fdb85)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #02 pc 0000000000006f14 /system/lib64/liblog.so (_android_log_assert+312) (BuildId:
467c2038cdfa767245f9280e657fdb85)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #03 pc 0000000000035294 /system/lib64/libmediautils.so
(android::TimeCheck::TimeCheckThread::threadLoop()+908) (BuildId: d1ac2ddd5dabadcca37691821970832)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #04 pc 0000000000013654 /system/lib64/libutils.so (android::Thread::_threadLoop(void)+464) (BuildId:
b940f836006eead456c936e97e64d2d4)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #05 pc 0000000000012de8 /system/lib64/libutils.so (thread_data_t::trampoline(thread_data_t const*)+408)
(BuildId: b940f836006eead456c936e97e64d2d4)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #06 pc 00000000000b6a24 /apex/com.android.runtime/lib64/bionic/libc.so (_pthread_start(void*)+264) (BuildId:
369edc656806aeaf384cbeb8f7a347af)
05-20 05:03:15.694 1041 30856 30856 F DEBUG : #07 pc 00000000000532bc /apex/com.android.runtime/lib64/bionic/libc.so (_start_thread+68) (BuildId:
369edc656806aeaf384cbeb8f7a347af)

```

Youku:

```

05-20 01:33:57.142 10105 32503 1430 F libc : Fatal signal 7 (SIGBUS), code 1 (BUS_ADRALN), fault addr 0xbca1c312 in tid 1430 (NativeThread), pid 32503
(com.youku.phone)
05-20 01:34:06.249 10105 1947 1947 F DEBUG : *
05-20 01:34:06.249 10105 1947 1947 F DEBUG : Build fingerprint: 'HiPhi/PassengerDisplay_CN/PassengerDisplay:12/SKQ1.220201.001/3388:user/test-keys'
05-20 01:34:06.249 10105 1947 1947 F DEBUG : Revision: '0'
05-20 01:34:06.249 10105 1947 1947 F DEBUG : ABI: 'arm'
05-20 01:34:06.249 10105 1947 1947 F DEBUG : Timestamp: 2023-05-20 01:33:58.263847092+0800
05-20 01:34:06.250 10105 1947 1947 F DEBUG : Process uptime: 0s
05-20 01:34:06.250 10105 1947 1947 F DEBUG : Cmdline: com.youku.phone
05-20 01:34:06.250 10105 1947 1947 F DEBUG : pid: 32503, tid: 1430, name: NativeThread >>> com.youku.phone <<<
05-20 01:34:06.250 10105 1947 1947 F DEBUG : uid: 10105
05-20 01:34:06.250 10105 1947 1947 F DEBUG : signal 7 (SIGBUS), code 1 (BUS_ADRALN), fault addr 0xbca1c312
05-20 01:34:06.250 10105 1947 1947 F DEBUG : r0 b9eade70 r1 bca1c312 r2 00000000 r3 00000000
05-20 01:34:06.250 10105 1947 1947 F DEBUG : r4 b9eade70 r5 00000031 r6 b9eade90 r7 b9eaded0
05-20 01:34:06.250 10105 1947 1947 F DEBUG : r8 00000000 r9 f5181060 r10 00000000 r11 00000000
05-20 01:34:06.250 10105 1947 1947 F DEBUG : ip f207d1d0 sp b9eade48 lr f2a31ca7 pc f132065a
05-20 01:34:06.250 10105 1947 1947 F DEBUG : backtrace:
05-20 01:34:06.250 10105 1947 1947 F DEBUG : #00 pc 0001465a /system/lib/libmediadrm.so (android::DrmHal::createPlugin(unsigned char const, android::String8

```

```
const&)+810) (BuildId: a09142dabde5c71f42cad8e0554d4644)
05-20 01:34:06.250 10105 1947 1947 F DEBUG : #01 pc 00011e3d /system/lib/libmediandk.so (AMediaDrm_createByUUID+808) (BuildId:
61b96ce3f03e71f3fb954f85fb05b3e5)
05-20 01:34:06.250 10105 1947 1947 F DEBUG : #02 pc 000a371f
/data/app/~~Xevg4M49I0VVCDVEtZ9vZA==/com.youku.phone-d_69SLwfiSP_Xrsb6688sg==/lib/arm/libsgmainso-6.5.15319409.so
05-20 01:34:06.250 10105 1947 1947 F DEBUG : #03 pc 000a3d7b
/data/app/~~Xevg4M49I0VVCDVEtZ9vZA==/com.youku.phone-d_69SLwfiSP_Xrsb6688sg==/lib/arm/libsgmainso-6.5.15319409.so
05-20 01:34:06.250 10105 1947 1947 F DEBUG : #04 pc 000ae788
/data/app/~~Xevg4M49I0VVCDVEtZ9vZA==/com.youku.phone-d_69SLwfiSP_Xrsb6688sg==/lib/arm/libsgmainso-6.5.15319409.so
05-20 02:20:34.762 1017 578 578 F libc : crash_dump helper failed to exec, or was killed
```

2.从log中看到hht_ota_mgr_service一直在重启，会影响系统的性能，需要确认下hht_ota_mgr_service一直启动失败的原因

```
05-20 08:13:00.069 root 0 0 I [40182.436662] init: starting service 'hht_ota_mgr_service'...
05-20 08:13:00.137 root 0 0 W [40182.504708] audit: audit_lost=138801 audit_rate_limit=5 audit_backlog_limit=64
05-20 08:13:00.137 root 0 0 E [40182.504710] audit: rate limit exceeded
05-20 08:13:00.145 root 0 0 I [40182.515023] init: Service 'hht_ota_mgr_service' (pid 8474) exited with status 0
05-20 08:13:00.145 root 0 0 I [40182.515049] init: Sending signal 9 to service 'hht_ota_mgr_service' (pid 8474) process group...
05-20 08:13:00.145 root 0 0 I [40182.515197] libprocessgroup: Successfully killed process cgroup uid 0 pid 8474 in 0ms
```

3.关于system_server的android.display的Watchdog异常。

```
05-20 08:20:40.159 1000 1525 1633 E Watchdog: First set of traces taken from /data/anr/anr_2023-05-20-08-20-06-979
05-20 08:20:40.190 1000 1525 1633 E Watchdog: Second set of traces taken from /data/anr/anr_2023-05-20-08-20-30-494
05-20 08:20:40.226 1000 1525 15951 I DropBoxManagerService: add tag=system_server_watchdog isTagEnabled=true flags=0x6
05-20 08:20:40.242 1000 1525 1633 W Watchdog: * WATCHDOG KILLING SYSTEM PROCESS: Blocked in handler on display thread (android.display)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: android.display annotated stack trace:
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.nativePollOnce(Native Method)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.next(MessageQueue.java:335)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loopOnce(Looper.java:161)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loop(Looper.java:288)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at android.os.HandlerThread.run(HandlerThread.java:67)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at com.android.server.ServiceThread.run(ServiceThread.java:44)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: * GOODBYE!
```

从这个trace分析是android.display处理消息超过了1分钟。

但是又没有显示是被谁block的。

出现了黑屏和冻屏后，watchdog异常然后发生了。

#16 - 2023-05-26 09:35 - CD FW 郑雷

[当前状态]

关于system_server的android.display的Watchdog异常。

```
05-20 08:20:40.159 1000 1525 1633 E Watchdog: First set of traces taken from /data/anr/anr_2023-05-20-08-20-06-979
05-20 08:20:40.190 1000 1525 1633 E Watchdog: Second set of traces taken from /data/anr/anr_2023-05-20-08-20-30-494
```

```
05-20 08:20:40.226 1000 1525 15951 I DropBoxManagerService: add tag=system_server_watchdog isTagEnabled=true flags=0x6
05-20 08:20:40.242 1000 1525 1633 W Watchdog: * WATCHDOG KILLING SYSTEM PROCESS: Blocked in handler on display thread (android.display)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: android.display annotated stack trace:
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.nativePollOnce(Native Method)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.next(MessageQueue.java:335)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loopOnce(Looper.java:161)
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loop(Looper.java:288)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at android.os.HandlerThread.run(HandlerThread.java:67)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at com.android.server.ServiceThread.run(ServiceThread.java:44)
05-20 08:20:40.246 1000 1525 1633 W Watchdog: * GOODBYE!
```

从这个trace分析是android.display处理消息超过了1分钟。

但是没有显示是被谁block的。

继续分析system_server的主线程和android.display的调用堆栈：

"main" prio=5 tid=1 Native

```
| group="main" sCount=1 ucsCount=0 flags=1 obj=0x727464a8 self=0xb4000078cfb24be0
| sysTid=16187 nice=-2 cgrp=default sched=0/0 handle=0x7a93fd14f8
| state=S schedstat=( 1990442010 704058801 5597 ) utm=144 stm=54 core=2 HZ=100
| stack=0x7fe58a8000-0x7fe58aa000 stackSize=8188KB
| held mutexes=
native: #00 pc 00000000000a2c4c /apex/com.android.runtime/lib64/bionic/libc.so (__epoll_pwait+12)
native: #01 pc 0000000000017ea8 /system/lib64/libutils.so (android::Looper::pollInner(int)+184)
native: #02 pc 0000000000017d84 /system/lib64/libutils.so (android::Looper::pollOnce(int, int*, int*, void**)+116)
native: #03 pc 0000000000152d3c /system/lib64/libandroid_runtime.so (android::android_os_MessageQueue_nativePollOnce(_JNIEnv*, _jobject*, long, int)+48)
at android.os.MessageQueue.nativePollOnce(Native method)
at android.os.MessageQueue.next(MessageQueue.java:335)
at android.os.Looper.loopOnce(Looper.java:161)
at android.os.Looper.loop(Looper.java:288)
at com.android.server.SystemServer.run(SystemServer.java:909)
at com.android.server.SystemServer.main(SystemServer.java:616)
at java.lang.reflect.Method.invoke(Native method)
at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:548)
at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:987)
```

从主线程看是在执行native方法，没有异常。

"android.display" prio=5 tid=14 Native

```
| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c00970 self=0xb400007573f026a0
| sysTid=1630 nice=-3 cgrp=default sched=0/0 handle=0x7364bb5cb0
| state=S schedstat=( 840607104521 617104182572 3271685 ) utm=57132 stm=26928 core=6 HZ=100
| stack=0x7364ab2000-0x7364ab4000 stackSize=1039KB
| held mutexes=
native: #00 pc 000000000004def4 /apex/com.android.runtime/lib64/bionic/libc.so (syscall+36)
native: #01 pc 00000000000527bc /apex/com.android.runtime/lib64/bionic/libc.so (_futex_wait_ex(void volatile*, bool, int, bool, timespec const*)+148)
native: #02 pc 00000000000b5d20 /apex/com.android.runtime/lib64/bionic/libc.so (pthread_cond_timedwait+144)
native: #03 pc 0000000000056d48 /system/lib64/libc++.so (std::__1::condition_variable::__do_timed_wait(std::__1::unique_lock<std::__1::mutex>&,
std::__1::chrono::time_point<std::__1::chrono::system_clock, std::__1::chrono::duration<long long, std::__1::ratio<1l, 1000000000l> > >)+112)
native: #04 pc 0000000000084b84 /system/lib64/libgui.so (android::BufferQueueProducer::waitForFreeSlotThenRelock(android::BufferQueueProducer::FreeSlotCaller,
std::__1::unique_lock<std::__1::mutex>&, int*) const+824)
native: #05 pc 0000000000084e84 /system/lib64/libgui.so (android::BufferQueueProducer::dequeueBuffer(int*, android::sp<android::Fence>*, unsigned int, unsigned int,
int, unsigned long, unsigned long*, android::FrameEventHistoryDelta*)+624)
native: #06 pc 00000000000e0ff0 /system/lib64/libgui.so (android::Surface::dequeueBuffer(ANativeWindowBuffer**, int*)+396)
```

```
native: #07 pc 0000000000e8870 /system/lib64/libgui.so (android::Surface::lock(ANativeWindow_Buffer*, ARect*)+144)
native: #08 pc 00000000001564c /system/lib64/libinputservice.so (android::SpriteIcon::draw(android::sp<android::Surface>) const+84)
native: #09 pc 000000000013120 /system/lib64/libinputservice.so (android::SpriteController::doUpdateSprites()+1212)
native: #10 pc 000000000012bbc /system/lib64/libinputservice.so (android::SpriteController::handleMessage(android::Message const&)+60)
native: #11 pc 0000000000171f8 /system/lib64/libutils.so (android::WeakMessageHandler::handleMessage(android::Message const&)+108)
native: #12 pc 000000000017f64 /system/lib64/libutils.so (android::Looper::pollInner(int)+372)
native: #13 pc 000000000017d84 /system/lib64/libutils.so (android::Looper::pollOnce(int, int*, int*, void**)+116)
native: #14 pc 0000000000152d3c /system/lib64/libandroid_runtime.so (android::android_os_MessageQueue_nativePollOnce(_JNIEnv*, _jobject*, long, int)+48)
at android.os.MessageQueue.nativePollOnce(Native method)
at android.os.MessageQueue.next(MessageQueue.java:335)
at android.os.Looper.loopOnce(Looper.java:161)
at android.os.Looper.loop(Looper.java:288)
at android.os.HandlerThread.run(HandlerThread.java:67)
at com.android.server.ServiceThread.run(ServiceThread.java:44)
```

从android.display线程的堆栈中也没有被block的原因。
猜测android.display线程处理消息慢跟系统当时的性能有关系。

在bugreport中有很多binder的log：
其中1525是system_server进程，1015是surfaceflinger进程，677是netd进程。
在Watchdog发生之前，surfaceflinger进程和netd进程不停的在向systemserver发起IPC通信，造成了system_server的binder线程池没有空闲的线程，并且system_server的binder_buffer被耗尽。
这有可能会导app端anr或者system_server anr出现异常

```
untitled:19630: 05-20 08:19:51.399 root 0 0 I [40599.381619] binder_alloc: 1525: pid 1015 spamming oneway? 87 buffers allocated for a total size of 171000
untitled:19644: 05-20 08:19:51.431 root 0 0 I [40599.414407] binder_alloc: 1525: pid 1015 spamming oneway? 88 buffers allocated for a total size of 173304
untitled:19651: 05-20 08:19:51.450 root 0 0 I [40599.431686] binder_alloc: 1525: pid 1015 spamming oneway? 89 buffers allocated for a total size of 175608
untitled:19680: 05-20 08:19:51.929 root 0 0 I [40599.914474] binder_alloc: 1525: pid 677 spamming oneway? 1070 buffers allocated for a total size of 336656
untitled:19716: 05-20 08:19:58.067 root 0 0 I [40606.048333] binder_alloc: 1525: pid 677 spamming oneway? 1071 buffers allocated for a total size of 336920
untitled:19717: 05-20 08:19:59.223 root 0 0 I [40607.204280] binder_alloc: 1525: pid 677 spamming oneway? 1072 buffers allocated for a total size of 337168
untitled:19719: 05-20 08:20:00.067 root 0 0 I [40608.052097] binder_alloc: 1525: pid 677 spamming oneway? 1073 buffers allocated for a total size of 337432
untitled:20166: 05-20 08:20:01.858 root 0 0 I [40609.842448] binder_alloc: 1525: pid 677 spamming oneway? 1072 buffers allocated for a total size of 337248
untitled:20167: 05-20 08:20:01.858 root 0 0 I [40609.842535] binder_alloc: 1525: pid 677 spamming oneway? 1073 buffers allocated for a total size of 337520
untitled:20168: 05-20 08:20:01.863 root 0 0 I [40609.844259] binder_alloc: 1525: pid 677 spamming oneway? 1074 buffers allocated for a total size of 337784
untitled:20169: 05-20 08:20:01.875 root 0 0 I [40609.857901] binder_alloc: 1525: pid 677 spamming oneway? 1075 buffers allocated for a total size of 338080
untitled:20170: 05-20 08:20:01.875 root 0 0 I [40609.858043] binder_alloc: 1525: pid 677 spamming oneway? 1076 buffers allocated for a total size of 338352
untitled:20171: 05-20 08:20:01.919 root 0 0 I [40609.902994] binder_alloc: 1525: pid 677 spamming oneway? 1077 buffers allocated for a total size of 338608
untitled:20172: 05-20 08:20:01.959 root 0 0 I [40609.941230] binder_alloc: 1525: pid 677 spamming oneway? 1078 buffers allocated for a total size of 338864
untitled:20243: 05-20 08:20:07.075 root 0 0 I [40615.059677] binder_alloc: 1525: pid 677 spamming oneway? 1083 buffers allocated for a total size of 340248
untitled:20379: 05-20 08:20:07.143 root 0 0 I [40615.127194] binder_alloc: 1525: pid 677 spamming oneway? 1084 buffers allocated for a total size of 340592
untitled:20380: 05-20 08:20:07.143 root 0 0 I [40615.128213] binder_alloc: 1525: pid 677 spamming oneway? 1085 buffers allocated for a total size of 340864
untitled:20415: 05-20 08:20:08.075 root 0 0 I [40616.058084] binder_alloc: 1525: pid 677 spamming oneway? 1086 buffers allocated for a total size of 341128
untitled:20434: 05-20 08:20:08.161 root 0 0 I [40616.135262] binder_alloc: 1525: pid 1015 spamming oneway? 86 buffers allocated for a total size of 171288
untitled:20790: 05-20 08:20:08.289 root 0 0 I [40616.268313] binder_alloc: 1525: pid 677 spamming oneway? 1085 buffers allocated for a total size of 340848
untitled:20812: 05-20 08:20:08.347 root 0 0 I [40616.302021] binder_alloc: 1525: pid 677 spamming oneway? 1086 buffers allocated for a total size of 341168
untitled:20813: 05-20 08:20:08.369 root 0 0 I [40616.302070] binder_alloc: 1525: pid 677 spamming oneway? 1087 buffers allocated for a total size of 341544
untitled:20814: 05-20 08:20:08.369 root 0 0 I [40616.302108] binder_alloc: 1525: pid 677 spamming oneway? 1088 buffers allocated for a total size of 341816
untitled:20897: 05-20 08:20:16.867 root 0 0 I [40624.843718] binder_alloc: 1525: pid 677 spamming oneway? 1087 buffers allocated for a total size of 341656
untitled:21010: 05-20 08:20:09.214 root 0 0 I [40625.076127] binder_alloc: 1525: pid 1015 spamming oneway? 86 buffers allocated for a total size of 171936
untitled:21092: 05-20 08:20:14.650 root 0 0 I [40630.518914] binder_alloc: 1525: pid 677 spamming oneway? 1086 buffers allocated for a total size of 341400
untitled:21093: 05-20 08:20:14.650 root 0 0 I [40630.519062] binder_alloc: 1525: pid 677 spamming oneway? 1087 buffers allocated for a total size of 341672
untitled:21094: 05-20 08:20:14.654 root 0 0 I [40630.521505] binder_alloc: 1525: pid 677 spamming oneway? 1088 buffers allocated for a total size of 341944
```

```

untitled:21096: 05-20 08:20:16.214 root 0 0 I [40632.080997] binder_alloc: 1525: pid 677 spamming oneway? 1089 buffers allocated for a total size of 342208
untitled:21097: 05-20 08:20:16.659 root 0 0 I [40632.527433] binder_alloc: 1525: pid 677 spamming oneway? 1090 buffers allocated for a total size of 342456
untitled:21102: 05-20 08:20:18.218 root 0 0 I [40634.085031] binder_alloc: 1525: pid 677 spamming oneway? 1091 buffers allocated for a total size of 342720
untitled:21105: 05-20 08:20:18.758 root 0 0 I [40634.627732] binder_alloc: 1525: pid 677 spamming oneway? 1092 buffers allocated for a total size of 342968
untitled:21284: 05-20 08:20:19.634 root 0 0 I [40635.502119] binder_alloc: 1525: pid 677 spamming oneway? 1091 buffers allocated for a total size of 342912
untitled:21324: 05-20 08:20:19.667 root 0 0 I [40635.529692] binder_alloc: 1525: pid 677 spamming oneway? 1092 buffers allocated for a total size of 343168
untitled:21364: 05-20 08:20:19.694 root 0 0 I [40635.557616] binder_alloc: 1525: pid 677 spamming oneway? 1093 buffers allocated for a total size of 343432
untitled:21377: 05-20 08:20:19.794 root 0 0 I [40635.663998] binder_alloc: 1525: pid 677 spamming oneway? 1094 buffers allocated for a total size of 343744
untitled:21378: 05-20 08:20:19.795 root 0 0 I [40635.664079] binder_alloc: 1525: pid 677 spamming oneway? 1095 buffers allocated for a total size of 344016
untitled:21380: 05-20 08:20:19.808 root 0 0 I [40635.674694] binder_alloc: 1525: pid 677 spamming oneway? 1096 buffers allocated for a total size of 344384
untitled:21381: 05-20 08:20:19.808 root 0 0 I [40635.674785] binder_alloc: 1525: pid 677 spamming oneway? 1097 buffers allocated for a total size of 344656
untitled:21382: 05-20 08:20:19.808 root 0 0 I [40635.675475] binder_alloc: 1525: pid 677 spamming oneway? 1098 buffers allocated for a total size of 345048
untitled:21476: 05-20 08:20:26.226 root 0 0 I [40642.092858] binder_alloc: 1525: pid 677 spamming oneway? 1099 buffers allocated for a total size of 345312
untitled:21484: 05-20 08:20:27.382 root 0 0 I [40643.250275] binder_alloc: 1525: pid 677 spamming oneway? 1100 buffers allocated for a total size of 345552
untitled:21578: 05-20 08:20:30.436 root 0 0 I [40646.301717] binder_alloc: 1525: pid 677 spamming oneway? 1099 buffers allocated for a total size of 345296
untitled:21833: 05-20 08:20:30.494 root 0 0 I [40646.363505] binder_alloc: 1525: pid 1015 spamming oneway? 83 buffers allocated for a total size of 167616
untitled:21836: 05-20 08:20:30.509 root 0 0 I [40646.364858] binder_alloc: 1525: pid 677 spamming oneway? 1098 buffers allocated for a total size of 345360
untitled:21837: 05-20 08:20:30.509 root 0 0 I [40646.364952] binder_alloc: 1525: pid 677 spamming oneway? 1099 buffers allocated for a total size of 345632
untitled:21838: 05-20 08:20:30.509 root 0 0 I [40646.365410] binder_alloc: 1525: pid 677 spamming oneway? 1100 buffers allocated for a total size of 345936
untitled:21839: 05-20 08:20:30.509 root 0 0 I [40646.365545] binder_alloc: 1525: pid 677 spamming oneway? 1101 buffers allocated for a total size of 346208
untitled:21840: 05-20 08:20:30.509 root 0 0 I [40646.367295] binder_alloc: 1525: pid 677 spamming oneway? 1102 buffers allocated for a total size of 346480
untitled:22042: 05-20 08:20:30.666 root 0 0 I [40646.516131] binder_alloc: 1525: pid 677 spamming oneway? 1101 buffers allocated for a total size of 346280
untitled:65806: 05-20 07:33:36.469 root 0 0 I [37834.217283] binder_alloc: 1525: pid 1015 spamming oneway? 65 buffers allocated for a total size of 123808
untitled:65807: 05-20 07:33:36.523 root 0 0 I [37834.267772] binder_alloc: 1525: pid 1015 spamming oneway? 66 buffers allocated for a total size of 125992
untitled:65808: 05-20 07:33:36.603 root 0 0 I [37834.350992] binder_alloc: 1525: pid 1015 spamming oneway? 67 buffers allocated for a total size of 128176
untitled:65809: 05-20 07:33:36.619 root 0 0 I [37834.367716] binder_alloc: 1525: pid 1015 spamming oneway? 68 buffers allocated for a total size of 130360
untitled:65810: 05-20 07:33:36.656 root 0 0 I [37834.401254] binder_alloc: 1525: pid 1015 spamming oneway? 69 buffers allocated for a total size of 132544

```

续查看system_server的binder线程，

"Binder:1525_D" prio=5 tid=147 Blocked

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c1c048 self=0xb400007573ff24c0

| sysTid=2611 nice=0 cgrp=default sched=0/0 handle=0x72db9e0cb0

| state=S schedstat=(240820398847 295902349236 1739576) utm=16096 stm=7986 core=3 HZ=100

| stack=0x72db8e9000-0x72db8eb000 stackSize=991KB

| held mutexes=

at com.android.server.thunderperf.ThunderBoostPerfService\$1.updateResolution(ThunderBoostPerfService.java:109)

- waiting to lock <0x06e4d00f> (a java.lang.Object) held by thread 148

at com.thundercomm.perf.BoostPerfManager.updateResolution(BoostPerfManager.java:71)

at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1190)

at com.android.server.am.ActivityManagerService.startActivityAsUserWithFeature(ActivityManagerService.java:2956)

at android.app.IActivityManager\$Stub.onTransact(IActivityManager.java:4050)

at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:2528)

at android.os.Binder.execTransactInternal(Binder.java:1179)

at android.os.Binder.execTransact(Binder.java:1143)

"Binder:1525_E" prio=5 tid=148 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c1c2a8 self=0xb4000075740213b0

```

| sysTid=2619 nice=0 cgrp=default sched=0/0 handle=0x72db5e8cb0
| state=S schedstat=( 252656823101 303994184748 1808929 ) utm=16790 stm=8474 core=7 HZ=100
| stack=0x72db4f1000-0x72db4f3000 stackSize=991KB
| held mutexes=
native: #00 pc 000000000004def0 /apex/com.android.runtime/lib64/bionic/libc.so (syscall+32)
native: #01 pc 0000000000028dc74 /apex/com.android.art/lib64/libart.so (art::ConditionVariable::WaitHoldingLocks(art::Thread*)+152)
native: #02 pc 0000000000046edc0 /apex/com.android.art/lib64/libart.so (art::JNI<false>::NewObjectV(JNIEnv*, _jclass*, _jmethodID*, std::_va_list)+500)
native: #03 pc 0000000000046eb68 /apex/com.android.art/lib64/libart.so (art::JNI<false>::NewObject(JNIEnv*, _jclass*, _jmethodID*, ...)+124)
native: #04 pc 0000000000029394 /apex/com.android.art/lib64/libjavacore.so (Linux_open(JNIEnv*, _jobject*, _jstring*, int, int)+340)
at libcore.io.Linux.open(Native method)
at libcore.io.ForwardingOs.open(ForwardingOs.java:567)
at libcore.io.BlockGuardOs.open(BlockGuardOs.java:273)
at libcore.io.IoBridge.open(IoBridge.java:561)
at java.io.FileOutputStream.<init>(FileOutputStream.java:236)
at java.io.FileOutputStream.<init>(FileOutputStream.java:186)
at java.io.PrintWriter.<init>(PrintWriter.java:263)
at com.android.server.compat.CompatConfig.saveOverrides(CompatConfig.java:741)
- locked <0x084666f7> (a java.io.File)
at com.android.server.compat.CompatConfig.addOverrides(CompatConfig.java:263)
at com.android.server.compat.PlatformCompat.setOverrides(PlatformCompat.java:225)
at com.android.server.thunderperf.ThunderBoostPerfService.checkAndSetResolution(ThunderBoostPerfService.java:183)
at com.android.server.thunderperf.ThunderBoostPerfService.setResolution(ThunderBoostPerfService.java:117)
at com.android.server.thunderperf.ThunderBoostPerfService.access$300(ThunderBoostPerfService.java:52)
at com.android.server.thunderperf.ThunderBoostPerfService$1.updateResolution(ThunderBoostPerfService.java:110)
- locked <0x06e4d00f> (a java.lang.Object)
at com.thundercomm.perf.BoostPerfManager.updateResolution(BoostPerfManager.java:71)
at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1190)
at com.android.server.am.ActivityManagerService.startActivityAsUserWithFeature(ActivityManagerService.java:2956)
at android.app.IActivityManager$Stub.onTransact(IActivityManager.java:4050)
at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:2528)
at android.os.Binder.execTransactInternal(Binder.java:1179)
at android.os.Binder.execTransact(Binder.java:1143)

```

在startActivity的过程中，Binder:1525_D对进程16817（ThunderBoostPerfService）进行了binder调用。Binder:1525_D在调用ThunderBoostPerfService的方法时等待0x06e4d00f锁。

而0x06e4d00f锁又被Binder:1525_E（sysTid：16817）线程持有。

关于2619线程，在watchdog之前有很多addOverrideUnsafe相关的log，确认是否是不释放锁的原因。

```

05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.626 1000 1525 2191 I ActivityTaskManager: START u0 {act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] flg=0x10200000
cmp=com.ss.android.article.video.landscape/com.ss.android.article.video.activity.SplashActivity} from uid 2000
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.626 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.627 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.627 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.627 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.627 1000 1525 2609 I thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0

```

```

05-20 08:20:30.627 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.627 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.627 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.627 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.628 1000 1525 2609 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: PlatormCompat.setOverrides: Uid1=2000 Pid1=31616 Uid2=2000 Pid2=31616
05-20 08:20:30.632 1000 1525 2619 | thunderperf: PlatormCompat.setOverrides: needCheck:false
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.632 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.633 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.634 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.634 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.634 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.634 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.639 1000 1525 2191 | ActivityTaskManager: The Process com.netease.cloudmusic Already Exists in BG. So sending its PID: 14090
05-20 08:20:30.640 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.640 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.642 root 0 0 | [40646.510305] binder: 1015:1015 transaction failed 29201/-28, size 2164-32 line 3240
05-20 08:20:30.642 1000 1525 2191 | ActivityTaskManager: The Process com.netease.cloudmusic Already Exists in BG. So sending its PID: 14090
05-20 08:20:30.644 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.644 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.647 1000 1525 2020 | ActivityTaskManager: START u0 {act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] flg=0x10200000
cmp=com.android.settings/.Settings$RunningServicesActivity} from uid 2000
05-20 08:20:30.648 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.648 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: bef-state=0
05-20 08:20:30.649 1000 1525 2619 | thunderperf: CompatConfig.addOverrideUnsafe: aft-state=0

```

也可能是system_server的binder线程被block，system_server被hang住造成的watchdog。

[Next Action]

查看更多的log，定位问题的原因

#17 - 2023-05-26 15:24 - CD FW 郑雷

[当前状态]

根据Watchdog轮询的时间段：

05-20 08:19:40.575 1000 1525 1627 W Looper : Slow dispatch took 3617ms android.fg h=android.os.Handler
c=com.android.server.Watchdog\$HandlerChecker@3d66a07 m=0

.....

05-20 08:20:06.959 1000 1525 1633 I Watchdog: WAITED_HALF

.....

05-20 08:20:40.159 1000 1525 1633 E Watchdog: First set of traces taken from /data/anr/anr_2023-05-20-08-20-06-979

05-20 08:20:40.190 1000 1525 1633 E Watchdog: Second set of traces taken from /data/anr/anr_2023-05-20-08-20-30-494

05-20 08:20:40.226 1000 1525 15951 I DropBoxManagerService: add tag=system_server_watchdog isTagEnabled=true flags=0x6

05-20 08:20:40.242 1000 1525 1633 W Watchdog: *** WATCHDOG KILLING SYSTEM PROCESS: Blocked in handler on display thread (android.display)

05-20 08:20:40.245 1000 1525 1633 W Watchdog: android.display annotated stack trace:

查看这段时间内的app anr异常信息：

Subject: Input dispatching timed out (Application does not have a focused window)

----- pid 7833 at 2023-05-20 08:16:25.169260775+0800 -----

Cmd line: com.youku.phone

.....

"main" prio=5 tid=1 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x722a5278 self=0xe6ac4610

| sysTid=7833 nice=0 cgrp=default sched=0/0 handle=0xf59ab470

| state=S schedstat=(398089086 2084490325 602) utm=18 stm=21 core=2 HZ=100

| stack=0xff501000-0xff503000 stackSize=8188KB

| held mutexes=

native: #00 pc 000715a8 /apex/com.android.runtime/lib/bionic/libc.so (__ioctl+8)

native: #01 pc 0003f2c7 /apex/com.android.runtime/lib/bionic/libc.so (ioctl+26)

native: #02 pc 0003abff /system/lib/libbinder.so (android::IPCThreadState::talkWithDriver(bool)+226)

native: #03 pc 0003b9f1 /system/lib/libbinder.so (android::IPCThreadState::waitForResponse(android::Parcel*, int*)+124)

native: #04 pc 0003b747 /system/lib/libbinder.so (android::IPCThreadState::transact(int, unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+134)

native: #05 pc 00035f3b /system/lib/libbinder.so (android::BpBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+118)

native: #06 pc 000f0201 /system/lib/libandroid_runtime.so (android_os_BinderProxy_transact(_JNIEnv*, _jobject*, int, _jobject*, _jobject*, int)+84)

at android.os.BinderProxy.transactNative(Native method)

at android.os.BinderProxy.transact(BinderProxy.java:571)

at android.app.IActivityClientController\$Stub\$Proxy.getTaskForActivity(IActivityClientController.java:1708)

at android.app.ActivityClient.getTaskForActivity(ActivityClient.java:204)

at android.app.Activity.getTaskId(Activity.java:6794)

at com.youku.phone.ActivityWelcome.c(SourceFile:9)

at com.youku.phone.ActivityWelcome.onCreate(SourceFile:24)

at android.app.Activity.performCreate(Activity.java:8094)

```

at android.app.Activity.performCreate(Activity.java:8074)
at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1330)
at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3609)
at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3793)
at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:103)
at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:135)
at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2211)
at android.os.Handler.dispatchMessage(Handler.java:106)
at android.os.Looper.loopOnce(Looper.java:201)
at android.os.Looper.loop(Looper.java:288)
at android.app.ActivityThread.main(ActivityThread.java:7880)
at java.lang.reflect.Method.invoke(Native method)
at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:548)
at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1009)

```

出现ANR是因为com.youku.phone对wms进行跨进程的调用，wms没有返回。

没有返回的原因，结合前面的分析是因为wms没有申请到mGlobalLock锁

```

@Override
public int getTaskForActivity(IBinder token, boolean onlyRoot) {
    synchronized (mGlobalLock) {
        return ActivityRecord.getTaskForActivityLocked(token, onlyRoot);
    }
}

```

Subject: Input dispatching timed out (Application does not have a focused window)

----- pid 9134 at 2023-05-20 08:15:43.923923192+0800 -----

Cmd line: com.sohu.inputmethod.sogou.car

.....

"main" prio=5 tid=1 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x722a5278 self=0xe6ac4610

| sysTid=9134 nice=0 cgrp=default sched=0/0 handle=0xf59ab470

| state=S schedstat=(214457900 2132906928 376) utm=9 stm=11 core=1 HZ=100

| stack=0xff501000-0xff503000 stackSize=8188KB

| held mutexes=

native: #00 pc 000715a8 /apex/com.android.runtime/lib/bionic/libc.so (__ioctl+8)

native: #01 pc 0003f2c7 /apex/com.android.runtime/lib/bionic/libc.so (ioctl+26)

native: #02 pc 0003abff /system/lib/libbinder.so (android::IPCThreadState::talkWithDriver(bool)+226)

native: #03 pc 0003b999 /system/lib/libbinder.so (android::IPCThreadState::waitForResponse(android::Parcel*, int*)+36)

native: #04 pc 0003b747 /system/lib/libbinder.so (android::IPCThreadState::transact(int, unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+134)

native: #05 pc 00035f3b /system/lib/libbinder.so (android::BpBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+118)

native: #06 pc 000f0201 /system/lib/libandroid_runtime.so (android_os_BinderProxy_transact(_JNIEnv*, _jobject*, int, _jobject*, _jobject*, int)+84)

at android.os.BinderProxy.transactNative(Native method)

at android.os.BinderProxy.transact(BinderProxy.java:571)

at android.view.IWindowManager\$Stub\$Proxy.getCurrentAnimatorScale(IWindowManager.java:3812)

at android.view.WindowManagerGlobal.getWindowManagerService(WindowManagerGlobal.java:190)

- locked <0x0b41d831> (a java.lang.Class<android.view.WindowManagerGlobal>)

at android.view.WindowManagerGlobal.initialize(WindowManagerGlobal.java:168)

at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3788)

at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:103)

at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:135)

```
at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2211)
at android.os.Handler.dispatchMessage(Handler.java:106)
at android.os.Looper.loopOnce(Looper.java:201)
at android.os.Looper.loop(Looper.java:288)
at android.app.ActivityThread.main(ActivityThread.java:7880)
at java.lang.reflect.Method.invoke(Native method)
at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:548)
at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1009)
```

com.sohu.inputmethod.sogou.car 出现anr的原因，也是com.sohu.inputmethod.sogou.car对wms进行跨进程的调用。
wms在等待mGlobalLock锁

```
public float getCurrentAnimatorScale() {
    synchronized (mGlobalLock) {
        return mAnimationsDisabled ? 0 : mAnimatorDurationScaleSetting;
    }
}
```

再结合system_server的binder线程分析：

```
"Binder:1525_8" prio=5 tid=138 Blocked
| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c0c010 self=0xb40000757401a470
| sysTid=2191 nice=0 cgrp=default sched=0/0 handle=0x72e8c03cb0
| state=S schedstat=( 219257053196 260681178210 1574972 ) utm=14590 stm=7335 core=7 HZ=100
| stack=0x72e8b0c000-0x72e8b0e000 stackSize=991KB
| held mutexes=
at com.android.server.wm.ActivityTaskManagerService$LocalService.isGetTasksAllowed(ActivityTaskManagerService.java:5423)
- waiting to lock <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock) held by thread 135
at com.android.server.am.ActivityManagerService.getRunningAppProcesses(ActivityManagerService.java:8553)
at android.app.IActivityManager$Stub.onTransact(IActivityManager.java:3318)
at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:2528)
at android.os.Binder.execTransactInternal(Binder.java:1179)
at android.os.Binder.execTransact(Binder.java:1143)
```

"android.anim" , "Binder:1525_8" , "Binder:1525_A" , "Binder:1525_B" , "Binder:1525_C" , "Binder:1525_10" , "Binder:1525_13" , "Binder:1525_1B""Binder:1525_1E"
都在等待0x05476839锁。

该锁又被"Binder:1525_6"持有。

```
"Binder:1525_6" prio=5 tid=135 Native
| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c1afb0 self=0xb40000757400fd90
| sysTid=2020 nice=-10 cgrp=default sched=0/0 handle=0x72d072fcb0
| state=S schedstat=( 209663076616 249904276829 1495116 ) utm=13816 stm=7150 core=4 HZ=100
```

```

| stack=0x72d0638000-0x72d063a000 stackSize=991KB
| held mutexes=
native: #00 pc 000000000004def0 /apex/com.android.runtime/lib64/bionic/libc.so (syscall+32)
native: #01 pc 000000000028dc74 /apex/com.android.art/lib64/libart.so (art::ConditionVariable::WaitHoldingLocks(art::Thread*)+152)
native: #02 pc 000000000074cd0c /apex/com.android.art/lib64/libart.so (art::GoToRunnable(art::Thread*)+412)
native: #03 pc 000000000074cb38 /apex/com.android.art/lib64/libart.so (art::JniMethodEnd(unsigned int, art::Thread*)+28)
at com.qualcomm.qti.Performance.native_perf_hint(Native method)
at com.qualcomm.qti.Performance.perfHint(Performance.java:234)
at java.lang.reflect.Method.invoke(Native method)
at android.util.BoostFramework.perfHint(BoostFramework.java:368)
at com.android.server.wm.ActivityTaskSupervisor.startSpecificActivity(ActivityTaskSupervisor.java:1063)
at com.android.server.wm.EnsureActivitiesVisibleHelper.makeVisibleAndRestartIfNeeded(EnsureActivitiesVisibleHelper.java:222)
at com.android.server.wm.EnsureActivitiesVisibleHelper.setActivityVisibilityState(EnsureActivitiesVisibleHelper.java:149)
at com.android.server.wm.EnsureActivitiesVisibleHelper.lambda$process$0$EnsureActivitiesVisibleHelper(EnsureActivitiesVisibleHelper.java:100)
at com.android.server.wm.EnsureActivitiesVisibleHelper$$ExternalSyntheticLambda0.accept(unavailable:-1)
at com.android.server.wm.ActivityRecord.forAllActivities(ActivityRecord.java:4095)
at com.android.server.wm.WindowContainer.forAllActivities(WindowContainer.java:1457)
at com.android.server.wm.WindowContainer.forAllActivities(WindowContainer.java:1451)
at com.android.server.wm.EnsureActivitiesVisibleHelper.process(EnsureActivitiesVisibleHelper.java:99)
at com.android.server.wm.Task.lambda$ensureActivitiesVisible$20(Task.java:6073)
at com.android.server.wm.Task$$ExternalSyntheticLambda23.accept(unavailable:-1)
at com.android.server.wm.Task.forAllLeafTasks(Task.java:3920)
at com.android.server.wm.Task.ensureActivitiesVisible(Task.java:6073)
at com.android.server.wm.Task.ensureActivitiesVisible(Task.java:6050)
at com.android.server.wm.Task.goToSleepIfPossible(Task.java:5748)
at com.android.server.wm.RootWindowContainer.lambda$putTasksToSleep$28(RootWindowContainer.java:2896)
at com.android.server.wm.RootWindowContainer$$ExternalSyntheticLambda33.accept(unavailable:-1)
at com.android.server.wm.Task.forAllRootTasks(Task.java:3932)
at com.android.server.wm.WindowContainer.forAllRootTasks(WindowContainer.java:1770)
at com.android.server.wm.WindowContainer.forAllRootTasks(WindowContainer.java:1770)
at com.android.server.wm.WindowContainer.forAllRootTasks(WindowContainer.java:1770)
... repeated 3 times
at com.android.server.wm.WindowContainer.forAllRootTasks(WindowContainer.java:1763)
at com.android.server.wm.RootWindowContainer.putTasksToSleep(RootWindowContainer.java:2894)
at com.android.server.wm.ActivityTaskSupervisor.checkReadyForSleepLocked(ActivityTaskSupervisor.java:2003)
at com.android.server.wm.Task.checkReadyForSleep(Task.java:5694)
at com.android.server.wm.Task.resumeTopActivityUncheckedLocked(Task.java:6245)
at com.android.server.wm.RootWindowContainer.resumeFocusedTasksTopActivities(RootWindowContainer.java:2436)
at com.android.server.wm.ActivityStarter.resumeTargetRootTaskIfNeeded(ActivityStarter.java:2698)
at com.android.server.wm.ActivityStarter.recycleTask(ActivityStarter.java:2059)
at com.android.server.wm.ActivityStarter.startActivityInner(ActivityStarter.java:1719)
at com.android.server.wm.ActivityStarter.startActivityUnchecked(ActivityStarter.java:1576)
at com.android.server.wm.ActivityStarter.executeRequest(ActivityStarter.java:1183)
at com.android.server.wm.ActivityStarter.execute(ActivityStarter.java:671)
- locked <0x05476839> (a com.android.server.wm.WindowManagerGlobalLock)
at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1227)
at com.android.server.wm.ActivityTaskManagerService.startActivityAsUser(ActivityTaskManagerService.java:1199)
at com.android.server.am.ActivityManagerService.startActivityAsUserWithFeature(ActivityManagerService.java:2956)
at android.app.IActivityManager$Stub.onTransact(IActivityManager.java:4050)
at com.android.server.am.ActivityManagerService.onTransact(ActivityManagerService.java:2528)
at android.os.Binder.execTransactInternal(Binder.java:1179)
at android.os.Binder.execTransact(Binder.java:1143)

```

查看这两份log，持有锁0x05476839的线程是有变化，说明wms还是在工作的，所以考虑有可能是因为系统慢，android.display处理消息慢造成的？

05-20 08:20:40.159 1000 1525 1633 E Watchdog: First set of traces taken from /data/anr/anr_2023-05-20-08-20-06-979

05-20 08:20:40.190 1000 1525 1633 E Watchdog: Second set of traces taken from /data/anr/anr_2023-05-20-08-20-30-494

#18 - 2023-05-26 18:27 - CD FW 郑雷

[当前分析]

如何判断是不是系统慢

继续分析android.display线程的调用堆栈：

"android.display" prio=5 tid=14 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c00970 self=0xb400007573f026a0

| sysTid=1630 nice=-3 cgrp=default sched=0/0 handle=0x7364bb5cb0

| state=S schedstat=(840607104521 617104182572 3271685) utm=57132 stm=26928 core=6 HZ=100

| stack=0x7364ab2000-0x7364ab4000 stackSize=1039KB

| held mutexes=

native: #00 pc 000000000004def4 /apex/com.android.runtime/lib64/bionic/libc.so (syscall+36)

native: #01 pc 00000000000527bc /apex/com.android.runtime/lib64/bionic/libc.so (_futex_wait_ex(void volatile*, bool, int, bool, timespec const*)+148)

native: #02 pc 00000000000b5d20 /apex/com.android.runtime/lib64/bionic/libc.so (pthread_cond_timedwait+144)

native: #03 pc 0000000000056d48 /system/lib64/libc++.so (std::__1::condition_variable::__do_timed_wait(std::__1::unique_lock<std::__1::mutex>&, std::__1::chrono::time_point<std::__1::chrono::system_clock, std::__1::chrono::duration<long long, std::__1::ratio<1l, 1000000000l> > >)+112)

native: #04 pc 0000000000084b84 /system/lib64/libgui.so (android::BufferQueueProducer::waitForFreeSlotThenRelock(android::BufferQueueProducer::FreeSlotCaller, std::__1::unique_lock<std::__1::mutex>&, int*) const+824)

native: #05 pc 0000000000084e84 /system/lib64/libgui.so (android::BufferQueueProducer::dequeueBuffer(int*, android::sp<android::Fence>*, unsigned int, unsigned int, int, unsigned long, unsigned long*, android::FrameEventHistoryDelta*)+624)

native: #06 pc 00000000000e0ff0 /system/lib64/libgui.so (android::Surface::dequeueBuffer(ANativeWindowBuffer**, int*)+396)

native: #07 pc 00000000000e8870 /system/lib64/libgui.so (android::Surface::lock(ANativeWindow_Buffer*, ARect*)+144)

native: #08 pc 000000000001564c /system/lib64/libinputservice.so (android::SpriteIcon::draw(android::sp<android::Surface>) const+84)

native: #09 pc 0000000000013120 /system/lib64/libinputservice.so (android::SpriteController::doUpdateSprites()+1212)

native: #10 pc 0000000000012bbc /system/lib64/libinputservice.so (android::SpriteController::handleMessage(android::Message const&)+60)

native: #11 pc 00000000000171f8 /system/lib64/libutils.so (android::WeakMessageHandler::handleMessage(android::Message const&)+108)

native: #12 pc 0000000000017f64 /system/lib64/libutils.so (android::Looper::pollInner(int)+372)

native: #13 pc 0000000000017d84 /system/lib64/libutils.so (android::Looper::pollOnce(int, int*, int*, void**)+116)

native: #14 pc 0000000000152d3c /system/lib64/libandroid_runtime.so (android::android_os_MessageQueue_nativePollOnce(_JNIEnv*, _jobject*, long, int)+48)

at android.os.MessageQueue.nativePollOnce(Native method)

at android.os.MessageQueue.next(MessageQueue.java:335)

at android.os.Looper.loopOnce(Looper.java:161)

at android.os.Looper.loop(Looper.java:288)

at android.os.HandlerThread.run(HandlerThread.java:67)

at com.android.server.ServiceThread.run(ServiceThread.java:44)

watchdog检测到超时时会先打印线程的堆栈，从android.display线程的调用堆栈可以看到，android.display确实是在执行绘制的操作。

然后，watchdog线程会打印如下log,然后kill system_server进程。

```
05-20 08:20:40.242 1000 1525 1633 W Watchdog: * WATCHDOG KILLING SYSTEM PROCESS: Blocked in handler on display thread (android.display)
```

```
05-20 08:20:40.245 1000 1525 1633 W Watchdog: android.display annotated stack trace:
```

```
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.nativePollOnce(Native Method)
```

```
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.MessageQueue.next(MessageQueue.java:335)
```

```
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loopOnce(Looper.java:161)
```

```
05-20 08:20:40.245 1000 1525 1633 W Watchdog: at android.os.Looper.loop(Looper.java:288)
```

```
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at android.os.HandlerThread.run(HandlerThread.java:67)
```

```
05-20 08:20:40.246 1000 1525 1633 W Watchdog: at com.android.server.ServiceThread.run(ServiceThread.java:44)
```

```
05-20 08:20:40.246 1000 1525 1633 W Watchdog: * GOODBYE!
```

从Watchdog的这段log，看出android.display线程又是处于一个空闲的状态，说明android.display线程在watchdog打印android.display堆栈到系统关机这个时间段内，是在工作的，android.display线程并没有被block，只是处理消息比较慢。

关于系统慢的原因，主要有几点：

1. youku app一致在crash，会消耗系统内存，占用cpu。=====》需要确认下youku的问题
2. audioserver 在重启，会占用系统资源 =====》需要audioserver确认下这个问题
3. hht_ota_mgr_service一直在重启，会影响系统的性能 =====》需要客户确认下这个服务重启的原因
4. netd进程不停的在向system_server进程进行IPC通信，造成system_server的binder线程池和binder_buffer都耗尽，造成system_server异常，影响系统性能。=====》需要netd的同学确认下不停跟system_server通信的原因

#19 - 2023-05-27 18:19 - CD FW 郑雷

[问题总结]

1.关于watchdog exception是因为android.display线程在等待free buffer,超过60秒了还在wait，看门狗发生。

在WAITED_HALF时间点的android.display堆栈看，执行到android::BufferQueueProducer::waitForFreeSlotThenRelock()时在等待free buffer

```
05-20 08:20:06.959 1000 1525 1633 I Watchdog: WAITED_HALF
```

```
05-20 08:20:06.974 1000 1525 1633 I ActivityManager: dumpStackTraces pids=null nativepids=[577, 578, 1186, 1187, 676, 1188, 677, 936, 1160, 31945, 822, 534, 823, 1015, 824, 826, 1178, 859, 1243, 1276, 1212, 828, 27869, 27870]
```

```
05-20 08:20:06.979 1000 1525 1633 I ActivityManager: Dumping to /data/anr/anr_2023-05-20-08-20-06-979
```

"android.display" prio=5 tid=14 Native

| group="main" sCount=1 ucsCount=0 flags=1 obj=0x12c430b0 self=0xb400007573f026a0

| sysTid=1630 nice=-3 cgrp=default sched=0/0 handle=0x7364bb5cb0

| state=S schedstat=(840607067072 617104182572 3271683) utm=57132 stm=26928 core=0 HZ=100

| stack=0x7364ab2000-0x7364ab4000 stackSize=1039KB

| held mutexes=

native: #00 pc 000000000004def4 /apex/com.android.runtime/lib64/bionic/libc.so (syscall+36)

native: #01 pc 00000000000527bc /apex/com.android.runtime/lib64/bionic/libc.so (_futex_wait_ex(void volatile*, bool, int, bool, timespec const*)+148)

native: #02 pc 00000000000b5d20 /apex/com.android.runtime/lib64/bionic/libc.so (pthread_cond_timedwait+144)

```
native: #03 pc 000000000056d48 /system/lib64/libc++.so (std::__1::condition_variable::__do_timed_wait(std::__1::unique_lock<std::__1::mutex>&,
std::__1::chrono::time_point<std::__1::chrono::system_clock, std::__1::chrono::duration<long long, std::__1::ratio<1l, 1000000000l>>>)+112)
native: #04 pc 0000000000084b84 /system/lib64/libgui.so (android::BufferQueueProducer::waitForFreeSlotThenRelock(android::BufferQueueProducer::FreeSlotCaller,
std::__1::unique_lock<std::__1::mutex>&, int*) const+824)
native: #05 pc 0000000000084e84 /system/lib64/libgui.so (android::BufferQueueProducer::dequeueBuffer(int*, android::sp<android::Fence>*, unsigned int, unsigned int,
int, unsigned long, unsigned long*, android::FrameEventHistoryDelta*)+624)
native: #06 pc 00000000000e0ff0 /system/lib64/libgui.so (android::Surface::dequeueBuffer(ANativeWindowBuffer**, int*)+396)
native: #07 pc 00000000000e8870 /system/lib64/libgui.so (android::Surface::lock(ANativeWindow_Buffer*, ARect*)+144)
native: #08 pc 000000000001564c /system/lib64/libinputservice.so (android::SpriteIcon::draw(android::sp<android::Surface>) const+84)
native: #09 pc 0000000000013120 /system/lib64/libinputservice.so (android::SpriteController::doUpdateSprites()+1212)
native: #10 pc 0000000000012bbc /system/lib64/libinputservice.so (android::SpriteController::handleMessage(android::Message const&)+60)
native: #11 pc 00000000000171f8 /system/lib64/libutils.so (android::WeakMessageHandler::handleMessage(android::Message const&)+108)
native: #12 pc 0000000000017f64 /system/lib64/libutils.so (android::Looper::pollInner(int)+372)
native: #13 pc 0000000000017d84 /system/lib64/libutils.so (android::Looper::pollOnce(int, int*, int*, void**)+116)
native: #14 pc 0000000000152d3c /system/lib64/libandroid_runtime.so (android::android_os_MessageQueue_nativePollOnce(_JNIEnv*, _jobject*, long, int)+48)
at android.os.MessageQueue.nativePollOnce(Native method)
at android.os.MessageQueue.next(MessageQueue.java:335)
at android.os.Looper.loopOnce(Looper.java:161)
at android.os.Looper.loop(Looper.java:288)
at android.os.HandlerThread.run(HandlerThread.java:67)
at com.android.server.ServiceThread.run(ServiceThread.java:44)
```

Watchdog Exception 发生时，android.display的堆栈也显示执行到android::BufferQueueProducer::waitForFreeSlotThenRelock()时继续在等待free buffer.

从以上的分析，waitForFreeSlotThenRelock被block跟系统性能有关系，需要起高通case确认下waitForFreeSlotThenRelock是否有可优化的空间
高通case id : 06647322

#20 - 2023-05-30 16:22 - CD FW王武军

1、5月30日最新进展

1) 之前分析的角度有很多，大致原因归咎于系统变慢了

关于waitForFreeSlotThenRelock是否可以优化的问题，高通的case有了最新的回复，
他们认为waitForFreeSlotThenRelock是一个google issue，他们提供了一个google给的日志patch，
大致策略就是，需要把这个patch加入，继续跑测试，根据日志进行进一步的分析。
是否需要将这个patch加入，跑本地测试？

与高通的沟通：如果是google issue 是否有对应的issue号进行追踪，这个信息还在与高通确认中。

2) 自己的分析

目前的原因归咎于系统变慢，我目前想从这个角度展开，系统是从什么时候开始变慢的，

另外结合我之前的分析怀疑地：

在05-20 08:20 之前，前10分钟出现了大量的：

binder等待和monkey注入失败的日志：

05-20 08:18:52.061 1525 2080 E IPCThreadState: binder thread pool (31 threads) starved for 5213 ms

05-20 08:19:08.471 1525 1627 W IPCThreadState: Waiting for thread to be free. mExecutingThreadsCount=32 mMaxThreads=31

05-20 08:19:08.345 6893 6893 I Monkey : // Injection Failed

05-20 08:19:08.346 6893 6893 I Monkey : Sleeping for 500 milliseconds

所以我的思路是：

(1)既然出现binder饥饿和等待，那么可以从侧面说面系统确实慢，那么慢是从什么时候开始的？

(2)monkey既然出现了注入失败，那么这个失败是从什么时候开始的？会不会与系统变慢的开始有关系？

按照上面的两个想法，对日志从最初进行一步一步分析：

系统变慢的苗头出现大概在：

05-19 21:35:07.175 1161 2557 E IPCThreadState: binder thread pool (1 threads) starved for 219 ms

monkey的注入失败大概是从：

05-19 21:34:51.693 9628 9628 I Monkey : :Sending Touch (ACTION_DOWN): 0:(2328.0,724.0)

05-19 21:34:51.694 9628 9628 I Monkey : :Sending Touch (ACTION_UP): 0:(2320.9473,723.29596)

05-19 21:34:51.739 18293 18293 I Monkey : :Sending Touch (ACTION_DOWN): 0:(1275.0,373.0)

05-19 21:34:51.739 18293 18293 I Monkey : :Sending Touch (ACTION_POINTER_DOWN 1): 0:(1260.8373,380.25552)

05-19 21:34:51.796 31610 31610 I Monkey : :Switch:

05-19 21:34:52.912 31610 31610 I Monkey : Sleeping for 500 milliseconds

05-19 21:34:53.088 8142 8142 I Monkey : Sleeping for 500 milliseconds

05-19 21:34:53.588 8142 8142 I Monkey : :Sending Touch (ACTION_DOWN): 0:(457.0,911.0)

05-19 21:34:53.938 1525 2609 W InputManager: Input event injection from pid 9628 failed.

05-19 21:34:53.938 9628 9628 I Monkey : // Injection Failed

05-19 21:34:54.043 1525 20850 W InputManager: Input event injection from pid 31610 failed.

05-19 21:34:54.046 31610 31610 I Monkey : // Injection Failed

05-19 21:34:54.557 1525 2611 W InputManager: Input event injection from pid 8142 failed.

05-19 21:34:54.558 8142 8142 I Monkey : // Injection Failed

05-19 21:34:58.675 1525 26556 W InputManager: Input event injection from pid 18293 failed.

05-19 21:34:58.675 18293 18293 I Monkey : // Injection Failed

从上面的日志我们可以发现两点：

1) 注入失败与系统变慢的时间有某种联系；

2) monkey的注入失败，存在一个很奇怪的现象，系统中存在多个monkey的进程：9628、18293、8142、31610

意思就是失败的时候，系统有多个monkey进程在向input注入事件，但是都是失败的。理论上monkey的测试应该只会存在一个monkey的进程，为什么这个地方存在多个monkey进程，而且它们都在向系统注入事件？这个异常的现象会不会占用更多的系统资源，引起系统变慢？

2、下一步

1) 继续根据目前分析到的怀疑点，为什么会存在多个monkey的进程向系统注入事件？这个是否会有引起系统的变慢和异常？

2) 跟进高通的case沟通

1、5月31日最新进展

(1)、高通的patch已经提交并编译了Jenkins
http://10.0.76.24:8080/job/VerifyBuild_for_IOT_6490/4077/
<https://dev.thundercomm.com/gerrit/c/general/platform/frameworks/native/+/203611>

还未找测试进行复现和验证（没有一个明确的复现问题的方法），以抓取出现问题时日志提供给高通继续帮忙分析。

(2)、按照昨天的分析发现继续这个方向的分析

昨天发现在发生anr前存在monkey注入失败，并且存在多个monkey进程同时在注入事件的问题，
今天从发生anr之前最早出现注入失败到，发生anr时这个时间段的注入失败信息可以发现：

FSEM_000090_0135_0519220343
monkey进程：9628、31610、8142、18293、29376、12606、28656
FSEM_000090_0136_0519223342
monkey进程：31610、28656、9628、22020、8142、26167、5510、18293、22020、29376、12606、15475、9767
FSEM_000090_0138_0519233342
monkey进程：31610、28656、9628、22020、8142、26167、5510、18293、22020、29376、12606、15475、9767

FSEM_000090_0155_0520080420
05-20 07:36:14.030 21249 21249 I Monkey : // Injection Failed
05-20 07:36:15.810 10382 10382 I Monkey : // Injection Failed
05-20 07:36:16.610 22503 22503 I Monkey : // Injection Failed
05-20 07:36:17.610 29112 29112 I Monkey : // Injection Failed
05-20 07:36:18.411 23706 23706 I Monkey : // Injection Failed
05-20 07:36:19.210 16216 16216 I Monkey : // Injection Failed
05-20 07:36:20.210 26025 26025 I Monkey : // Injection Failed
05-20 07:36:21.810 17903 17903 I Monkey : // Injection Failed
05-20 07:36:24.010 23600 23600 I Monkey : // Injection Failed
05-20 07:36:24.610 8646 8646 I Monkey : // Injection Failed
05-20 07:36:26.210 5334 5334 I Monkey : // Injection Failed

05-20 07:36:27.011 21401 21401 I Monkey : // Injection Failed
05-20 07:36:28.810 5064 5064 I Monkey : // Injection Failed
05-20 07:36:29.610 20627 20627 I Monkey : // Injection Failed
05-20 07:36:30.611 29868 29868 I Monkey : // Injection Failed
05-20 07:36:31.011 11254 11254 I Monkey : // Injection Failed
05-20 07:36:31.410 19797 19797 I Monkey : // Injection Failed
05-20 07:36:31.811 18885 18885 I Monkey : // Injection Failed
05-20 07:36:31.814 19072 19072 I Monkey : // Injection Failed
05-20 07:36:31.814 17761 17761 I Monkey : // Injection Failed
05-20 07:36:31.817 5329 5329 I Monkey : // Injection Failed
05-20 07:36:31.817 31914 31914 I Monkey : // Injection Failed
05-20 07:36:31.817 28656 28656 I Monkey : // Injection Failed
05-20 07:36:31.817 32503 32503 I Monkey : // Injection Failed
05-20 07:36:31.818 6005 6005 I Monkey : // Injection Failed
05-20 07:36:31.818 6355 6355 I Monkey : // Injection Failed
05-20 07:36:31.818 18293 18293 I Monkey : // Injection Failed
05-20 07:36:31.819 26167 26167 I Monkey : // Injection Failed
05-20 07:36:31.819 9161 9161 I Monkey : // Injection Failed
05-20 07:36:31.819 1752 1752 I Monkey : // Injection Failed
05-20 07:36:31.820 32486 32486 I Monkey : // Injection Failed
05-20 07:36:31.820 24810 24810 I Monkey : // Injection Failed
05-20 07:36:31.820 16030 16030 I Monkey : // Injection Failed
05-20 07:36:31.820 5530 5530 I Monkey : // Injection Failed
05-20 07:36:31.821 4595 4595 I Monkey : // Injection Failed
05-20 07:36:31.821 22705 22705 I Monkey : // Injection Failed
05-20 07:36:31.821 32025 32025 I Monkey : // Injection Failed
05-20 07:36:31.822 5178 5178 I Monkey : // Injection Failed
05-20 07:36:31.822 25410 25410 I Monkey : // Injection Failed
05-20 07:36:31.822 16972 16972 I Monkey : // Injection Failed
05-20 07:36:31.824 19750 19750 I Monkey : // Injection Failed
05-20 07:36:31.824 2725 2725 I Monkey : // Injection Failed
05-20 07:36:31.825 11620 11620 I Monkey : // Injection Failed
05-20 07:36:31.825 29669 29669 I Monkey : // Injection Failed
05-20 07:36:31.825 8319 8319 I Monkey : // Injection Failed
05-20 07:36:31.825 26503 26503 I Monkey : // Injection Failed
05-20 07:36:31.825 13378 13378 I Monkey : // Injection Failed
05-20 07:36:31.826 25576 25576 I Monkey : // Injection Failed
05-20 07:36:31.826 10053 10053 I Monkey : // Injection Failed
05-20 07:36:31.827 28961 28961 I Monkey : // Injection Failed
05-20 07:36:31.827 8339 8339 I Monkey : // Injection Failed
05-20 07:36:31.828 8475 8475 I Monkey : // Injection Failed
05-20 07:36:31.828 28564 28564 I Monkey : // Injection Failed
05-20 07:36:31.828 31101 31101 I Monkey : // Injection Failed
05-20 07:36:31.828 11826 11826 I Monkey : // Injection Failed
05-20 07:36:31.829 8642 8642 I Monkey : // Injection Failed
05-20 07:36:31.830 32561 32561 I Monkey : // Injection Failed
05-20 07:36:31.830 5510 5510 I Monkey : // Injection Failed
05-20 07:36:31.831 17830 17830 I Monkey : // Injection Failed
05-20 07:36:31.831 17590 17590 I Monkey : // Injection Failed
05-20 07:36:31.831 12606 12606 I Monkey : // Injection Failed
05-20 07:36:31.831 4953 4953 I Monkey : // Injection Failed
05-20 07:36:31.831 18960 18960 I Monkey : // Injection Failed
05-20 07:36:31.831 11215 11215 I Monkey : // Injection Failed

05-20 07:36:31.832 13577 13577 I Monkey : // Injection Failed
05-20 07:36:31.832 27494 27494 I Monkey : // Injection Failed
05-20 07:36:31.832 757 757 I Monkey : // Injection Failed
05-20 07:36:31.832 16883 16883 I Monkey : // Injection Failed
05-20 07:36:31.833 31610 31610 I Monkey : // Injection Failed
05-20 07:36:31.833 26026 26026 I Monkey : // Injection Failed
05-20 07:36:31.833 12681 12681 I Monkey : // Injection Failed
05-20 07:36:31.834 29376 29376 I Monkey : // Injection Failed
05-20 07:36:31.834 30647 30647 I Monkey : // Injection Failed
05-20 07:36:31.835 14822 14822 I Monkey : // Injection Failed
05-20 07:36:31.835 8141 8141 I Monkey : // Injection Failed
05-20 07:36:31.835 27501 27501 I Monkey : // Injection Failed
05-20 07:36:31.835 19170 19170 I Monkey : // Injection Failed
05-20 07:36:31.836 15642 15642 I Monkey : // Injection Failed
05-20 07:36:31.836 27455 27455 I Monkey : // Injection Failed
05-20 07:36:31.836 17867 17867 I Monkey : // Injection Failed
05-20 07:36:31.836 8887 8887 I Monkey : // Injection Failed
05-20 07:36:31.837 18251 18251 I Monkey : // Injection Failed

FSEM_000090_0156_0520082113

05-20 08:12:52.021 25405 25405 I Monkey : // Injection Failed
05-20 08:12:52.022 31751 31751 I Monkey : // Injection Failed
05-20 08:12:52.022 22503 22503 I Monkey : // Injection Failed
05-20 08:12:52.023 18251 18251 I Monkey : // Injection Failed
05-20 08:12:52.023 22280 22280 I Monkey : // Injection Failed
05-20 08:12:52.023 22020 22020 I Monkey : // Injection Failed
05-20 08:12:52.023 16883 16883 I Monkey : // Injection Failed
05-20 08:12:52.023 29943 29943 I Monkey : // Injection Failed
05-20 08:12:52.023 27494 27494 I Monkey : // Injection Failed
05-20 08:12:52.024 5053 5053 I Monkey : // Injection Failed
05-20 08:12:52.024 20309 20309 I Monkey : // Injection Failed
05-20 08:12:52.024 23149 23149 I Monkey : // Injection Failed
05-20 08:12:52.024 5329 5329 I Monkey : // Injection Failed
05-20 08:12:52.024 31372 31372 I Monkey : // Injection Failed
05-20 08:12:52.024 26977 26977 I Monkey : // Injection Failed
05-20 08:12:52.024 1752 1752 I Monkey : // Injection Failed
05-20 08:12:52.025 12606 12606 I Monkey : // Injection Failed
05-20 08:12:52.025 12945 12945 I Monkey : // Injection Failed
05-20 08:12:52.025 24810 24810 I Monkey : // Injection Failed
05-20 08:12:52.025 19797 19797 I Monkey : // Injection Failed
05-20 08:12:52.025 27045 27045 I Monkey : // Injection Failed
05-20 08:12:52.025 16947 16947 I Monkey : // Injection Failed
05-20 08:12:56.766 8595 8595 I Monkey : // Injection Failed
05-20 08:12:56.766 26025 26025 I Monkey : // Injection Failed
05-20 08:12:56.766 19750 19750 I Monkey : // Injection Failed
05-20 08:12:56.766 11593 11593 I Monkey : // Injection Failed
05-20 08:12:56.766 18293 18293 I Monkey : // Injection Failed
05-20 08:12:56.766 32486 32486 I Monkey : // Injection Failed
05-20 08:12:56.767 11215 11215 I Monkey : // Injection Failed
05-20 08:12:56.767 18960 18960 I Monkey : // Injection Failed

小结：
可以发现，到发生anr时已经不止几个monkey进程在注入事件了，而是几十个monkey进程在注入事件，而且注入的所有事件都是失败，从这个层面来看，几十个monkey进程同时在向input注入事件，这必然会过度消耗系统的资源，引起anr或者系统的卡顿将会是必然结果。

2、下一步

1) 找到存在这么多monkey进程的原因，为什么会有这个现象，这是引发票中问题的一个直接原因？

#22 - 2023-06-01 17:08 - CD FW王武军

1、6月1日最新进展

1) 继续昨天的调查方向，确认为什么会存在几十个monkey的进程

今天从发生anr时，追溯到该次开机的最早日志，可以发现如下日志：

05-19 21:03:37.332 3996 3996 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:03:37.569 4023 4023 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:05:39.016 9628 9628 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:10:39.597 29376 29376 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:15:39.558 18293 18293 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:25:39.906 8142 8142 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:30:40.377 31610 31610 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 21:20:39.637 12606 12606 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:10:49.237 5329 5329 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:05:41.530 26025 26025 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:15:42.021 19072 19072 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:20:42.367 3734 3734 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:30:43.616 29112 29112 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:40:44.055 22503 22503 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:50:43.204 12945 12945 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:45:46.212 16398 16398 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:00:44.936 8436 8436 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:55:43.115 11826 11826 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 22:35:44.068 27494 27494 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:20:45.384 26977 26977 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<

05-19 23:25:45.486 31751 31751 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:30:45.452 6005 6005 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:05:46.253 8646 8646 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:15:45.360 21249 21249 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:45:57.202 22705 22705 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:00:45.994 7122 7122 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:50:46.622 27045 27045 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:40:58.200 16947 16947 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:55:51.658 31914 31914 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-19 23:36:26.589 13577 13577 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:20:47.099 1752 1752 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:15:47.007 26026 26026 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:30:47.169 16216 16216 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:25:47.115 8319 8319 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:05:48.883 13378 13378 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:40:47.239 30485 30485 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:46:03.447 5064 5064 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:35:47.252 23706 23706 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:00:47.247 23149 23149 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:51:24.865 10382 10382 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 00:55:47.410 15642 15642 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:15:41.790 5178 5178 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:21:34.959 10077 10077 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:30:47.974 26503 26503 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:26:39.304 18885 18885 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:06:05.051 27501 27501 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:11:38.229 31371 31371 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:57:33.202 18251 18251 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:01:29.245 24810 24810 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 01:50:51.633 7876 7876 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:20:47.091 9911 9911 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:26:10.857 21401 21401 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:30:48.613 30647 30647 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:45:48.926 27455 27455 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:50:48.982 5934 5934 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:01:19.023 18960 18960 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:40:48.850 17616 17616 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 02:55:53.744 12908 12908 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:15:50.355 8141 8141 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:26:06.555 32503 32503 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:30:50.573 17867 17867 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:45:50.524 19750 19750 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:40:55.599 8475 8475 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:00:51.567 26626 26626 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:55:47.316 19797 19797 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 03:50:50.360 5334 5334 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:15:50.134 17830 17830 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:31:05.494 10053 10053 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:25:41.659 1473 1473 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:05:49.665 4595 4595 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:21:06.191 25836 25836 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:10:42.344 11215 11215 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:35:49.145 17590 17590 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<

05-20 04:45:55.488 31616 31616 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:40:51.849 26601 26601 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:00:55.140 26180 26180 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 04:51:03.650 8642 8642 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:21:05.826 32561 32561 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:10:52.607 17903 17903 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:26:05.176 11254 11254 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:05:50.243 6355 6355 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:15:54.694 23600 23600 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:30:55.706 22166 22166 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:45:44.670 5530 5530 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:51:07.178 8887 8887 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:40:56.834 32486 32486 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:55:34.404 12681 12681 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 05:36:11.398 28564 28564 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:01:37.232 14822 14822 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:10:41.244 21946 21946 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:30:55.115 5053 5053 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:05:32.805 17761 17761 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:15:57.213 25576 25576 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:25:52.361 757 757 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:20:50.720 29669 29669 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:50:50.530 16883 16883 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:41:22.981 11620 11620 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:01:30.197 20993 20993 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:45:51.675 14311 14311 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:56:00.905 19170 19170 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 06:35:45.060 8339 8339 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:15:59.214 28961 28961 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:20:53.320 32502 32502 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:25:46.929 4953 4953 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:05:55.234 22859 22859 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:31:35.849 8595 8595 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:10:54.405 25410 25410 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:46:53.676 18841 18841 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:55:32.265 25405 25405 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:36:04.677 11593 11593 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:40:52.484 15756 15756 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:00:24.769 29943 29943 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 07:50:55.612 22280 22280 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:05:22.645 1746 1746 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:19:08.897 13644 13644 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:10:33.882 6893 6893 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:20:09.281 14537 14537 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:19:30.196 13945 13945 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:20:30.780 14861 14861 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:15:33.566 11285 11285 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:21:34.159 20445 20445 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
05-20 08:21:34.324 20473 20473 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<

可以看见到anr之前已经创建了120多个monkey的进程了，启动的日志基本如下：

```
1813: 05-19 21:05:39.016 9628 9628 D AndroidRuntime: >>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<
1814: 05-19 21:05:39.020 9628 9628 I AndroidRuntime: Using default boot image
1815: 05-19 21:05:39.020 9628 9628 I AndroidRuntime: Leaving lock profiling enabled
1816: 05-19 21:05:39.020 9628 9628 W libc : Access denied finding property "odsign.verificaton.success"
1817: 05-19 21:05:39.015 9628 9628 I auditd : type=1400 audit(0.0:3984): avc: denied { read } for comm="app_process" name="u:object_r:odsign_prop:s0" dev="tmpfs"
ino=12133 scontext=u:r:shell:s0 tcontext=u:object_r:odsign_prop:s0 tclass=file permissive=0
1818: 05-19 21:05:39.015 9628 9628 W app_process: type=1400 audit(0.0:3984): avc: denied { read } for name="u:object_r:odsign_prop:s0" dev="tmpfs" ino=12133
scontext=u:r:shell:s0 tcontext=u:object_r:odsign_prop:s0 tclass=file permissive=0
1819: 05-19 21:05:39.022 9628 9628 W app_process: ART APEX data files are untrusted.
 4: 05-19 21:05:39.137 9628 9628 D app_process: Time zone APEX ICU file found: /apex/com.android.tzdata/etc/icu/icu_tzdata.dat
 5: 05-19 21:05:39.137 9628 9628 D app_process: I18n APEX ICU file found: /apex/com.android.i18n/etc/icu/icudt68l.dat
 6: 05-19 21:05:39.173 9628 9628 D Build : ro.build.display.id=01.00.0100.X103
 7: 05-19 21:05:39.173 9628 9628 D ProcessState: Binder ioctl to enable oneway spam detection failed: Invalid argument
 8: 05-19 21:05:39.173 9628 9628 D AndroidRuntime: Calling main entry com.android.commands.monkey.Monkey
 9: 05-19 21:05:39.174 9628 9628 W Monkey : args: [--pkg-blacklist-file, /sdcard/blackList.txt, --hprof, --throttle, 500, -s, 1234567, --pct-touch, 35, --pct-motion, 25,
--pct-appswitch, 30, --pct-pinchzoom, 5, --pct-trackball, 0, --pct-anyevent, 5, --ignore-crashes, --ignore-timeouts, --ignore-security-exceptions, --monitor-native-crashes,
--pct-syskeys, 0, -v, -v, -v, 12000000]
10: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pkg-blacklist-file"
11: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "/sdcard/blackList.txt"
12: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--hprof"
13: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--throttle"
14: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "500"
15: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "-s"
16: 05-19 21:05:39.174 0 0 I [ 138.179163] binder: 9628:9628 ioctl 40046210 7fdaaf4644 returned -22
17: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "1234567"
18: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-touch"
19: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "35"
20: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-motion"
21: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "25"
22: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-appswitch"
23: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "30"
24: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-pinchzoom"
25: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "5"
26: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-trackball"
27: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "0"
28: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "--pct-anyevent"
29: 05-19 21:05:39.174 9628 9628 W Monkey : arg: "5"
30: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "--ignore-crashes"
31: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "--ignore-timeouts"
32: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "--ignore-security-exceptions"
33: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "--monitor-native-crashes"
34: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "--pct-syskeys"
35: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "0"
36: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "-v"
37: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "-v"
38: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "-v"
39: 05-19 21:05:39.175 9628 9628 W Monkey : arg: "12000000"
40: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pkg-blacklist-file" mCurArgData="null" mNextArg=1 argwas="--pkg-blacklist-file"
```

```
nextarg="/sdcard/blackList.txt"
41: 05-19 21:05:39.175 9628 9628 W Monkey : data="/sdcard/blackList.txt"
42: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--hprof" mCurArgData="null" mNextArg=3 argwas="--hprof" nextarg="--throttle"
43: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--throttle" mCurArgData="null" mNextArg=4 argwas="--throttle" nextarg="500"
44: 05-19 21:05:39.175 9628 9628 W Monkey : data="500"
45: 05-19 21:05:39.175 9628 9628 W Monkey : data="1234567"
46: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pct-touch" mCurArgData="null" mNextArg=8 argwas="--pct-touch" nextarg="35"
47: 05-19 21:05:39.175 9628 9628 W Monkey : data="35"
48: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pct-motion" mCurArgData="null" mNextArg=10 argwas="--pct-motion" nextarg="25"
49: 05-19 21:05:39.175 9628 9628 W Monkey : data="25"
50: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pct-appswitch" mCurArgData="null" mNextArg=12 argwas="--pct-appswitch" nextarg="30"
51: 05-19 21:05:39.175 9628 9628 W Monkey : data="30"
52: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pct-pinchzoom" mCurArgData="null" mNextArg=14 argwas="--pct-pinchzoom" nextarg="5"
53: 05-19 21:05:39.175 9628 9628 W Monkey : data="5"
54: 05-19 21:05:39.175 9628 9628 W Monkey : arg="--pct-trackball" mCurArgData="null" mNextArg=16 argwas="--pct-trackball" nextarg="0"
55: 05-19 21:05:39.176 9628 9628 W Monkey : data="0"
56: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--pct-anyevent" mCurArgData="null" mNextArg=18 argwas="--pct-anyevent" nextarg="5"
57: 05-19 21:05:39.176 9628 9628 W Monkey : data="5"
58: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--ignore-crashes" mCurArgData="null" mNextArg=20 argwas="--ignore-crashes" nextarg="--ignore-timeouts"
59: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--ignore-timeouts" mCurArgData="null" mNextArg=21 argwas="--ignore-timeouts"
nextarg="--ignore-security-exceptions"
60: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--ignore-security-exceptions" mCurArgData="null" mNextArg=22 argwas="--ignore-security-exceptions"
nextarg="--monitor-native-crashes"
61: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--monitor-native-crashes" mCurArgData="null" mNextArg=23 argwas="--monitor-native-crashes"
nextarg="--pct-syskeys"
62: 05-19 21:05:39.176 9628 9628 W Monkey : arg="--pct-syskeys" mCurArgData="null" mNextArg=24 argwas="--pct-syskeys" nextarg="0"
63: 05-19 21:05:39.176 9628 9628 W Monkey : data="0"
```

小结：

现在可以确定的信息：

1) 在整个11小时的monkey测试中，monkey进程创建了120多个，而且它们都会向系统注入事件，这个必然会引起系统响应问题。

2) 我自己本地做了测试，同时执行5条adb shell monkey的命令，系统中会存在5个monkey的进程，在5个monkey跑的过程中，系统很快出现anr，界面切换很卡顿。

不能确定的：

1) 这120多个monkey进程被启动的原因是什么？

我简单看了下monkey的源码没看到代码中自动启动的逻辑；

与测试沟通，他说他没有重复多次的执行monkey命令

2、下一步

1) 查找monkey被无限创建的原因？

#23 - 2023-06-02 19:32 - CD FW王武军

1、6月2日最新进展

(1) 追查monkey被多次启动的原因

1) 分析了monkey的源码没有发现：在monkey进程存在的情况，可以主动启动monkey的逻辑；

2) 从日志层面找相关的信息，也没有找到根本原因。只发现了（com.cmcc.cmvideo 咪咕视频）在整个monkey测试过程中有重启进程，并打印有关monkey的日志，

但是也没有办法确认monkey的启动与com.cmcc.cmvideo 咪咕视频的进程重启有直接关系。

(2) 该问题的目前小结

冻屏问题发生时出现system_server的anr。系统层面在发生anr的那一段时间，binder进程存在挂起和阻塞的现象。

从这个日志层面来看，系统出现卡，binder阻塞、进而system_server发生anr，应该都与monkey进程有关系。

从整个长达11小时的monkey测试来看，从monkey测试开始后的一段时间（05-19 21:03:37.569 4023 D AndroidRuntime: >>>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<）monkey开始被启动，一直持续到发生anr的时间点（05-20 08:19:30.196 13945 13945 D AndroidRuntime: >>>>>> START com.android.internal.os.RuntimeInit uid 2000 <<<<<<）

最后从11小时的日志中，统计下来大概monkey进程创建了122个，这些进程大多都没有被销毁，从这个角度来看有122多个monkey进程在向input注入事件，这必然会引起系统的卡顿和资源的消耗，从日志中也都有体现，不管是IPCThreadState一直在报进程的饥饿和等待，还是monkey的事件一直没有被注入成功的报错。

所以整体来看，我们根据现在的分析大致可以得出一个结论：系统出现冻屏，直接原因应该是与monkey的进程一直被创建，monkey的事件注入操作占用大量的binder资源和系统资源。

目前的处理策略：
1) monkey一直被启动的根本原因未能找到，只能从系统层面进行检测，拦截monkey被无限创建的可能
初步方案的提交如下：
<https://dev.thundercomm.com/gerrit/c/general/platform/development/+/204393>

2、下一步

1) 本地验证拦截monkey多起启动的可行性和周边，后续确认拦截方案是否可以入库。

#24 - 2023-06-05 15:20 - IoT scm
Gerrit Merge Information :

ID	Project	Branch	Uploader
204393	general/platform/development	Pre_figure_turbox-c2130c-la1.1-qssi12-dev	wujun.wang@thundersoft.com
FW:SystemOptimization: intercept more monkey testAdd logic to intercept more monkey process to run.TC-RID: 1201-0205101IssueID:			

#25 - 2023-06-05 15:42 - CD FW王武军

6月5日最新进展

1、目前从日志中分析出的3个主要问题

1) youku crash的问题

audio组已经在子票118237中做了规避方案。

2) 关于BufferQueueProducer::waitForFreeSlotThenRelock()等待的问题

起了高通case，高通通过google给了一个打印日志的patch，添加patch编译vb版本，测试并未复现相同问题。

3) monkey进程重复创建问题

目前做了一个规避方案，避免monkey进程重复创建，提交已经入库。

2、下一步

该票中根据日志分析出的3个主要问题都进行了针对性的处理，后续策略是走票，然后同事进行针对性的测试，看是否还会存在类似的问题。

#26 - 2023-06-05 15:47 - CD FW王武军

- Status changed from ASSIGNED to RESOLVED

- Assignee changed from CD FW 郑雷 to CDTS_TEST 王成

- Resolution changed from -- to FIXED

- Root cause set to 问题原因：

问题的表现是黑屏和冻屏，但是引起的原因，大致如票中的分析信息

问题处理：

针对日志中已经发现的可能引起黑屏和冻屏的问题，进行了处理和验证

#27 - 2023-06-19 16:28 - CD TEST-方永红

6.19

五台设备未复现黑屏和冻屏

#28 - 2023-06-25 17:39 - CDTS_TEST 王成

- Assignee changed from CDTS_TEST 王成 to CD TEST-方永红

持续跟踪压测

#29 - 2023-07-25 16:21 - CD TEST-方永红
- Status changed from RESOLVED to VERIFIED

7.25
7月份的测试中，未出现monkey后黑屏的问题，先关闭

#30 - 2023-07-25 16:21 - CD TEST-方永红
- Status changed from VERIFIED to CLOSED