

Figure - Bug #118327

【AF】 【EVT3】 【Monkey】 Device#2(4+27个APK（无豌豆荚）-内部测试命令),异常终止

2023-05-29 10:09 - CD TEST-方永红

Status:	CLOSED	Start date:	2023-05-18
Priority:	Urgent	Due date:	2023-06-05
Assignee:	CD TEST-方永红	% Done:	100%
Category:	CD-FW	Estimated time:	0.00 hour
Target version:			
Need_Info:	--	Found Version:	01.00.0105.C104
Resolution:	FIXED	Degrated:	No
Severity:	Critical	Verified Version:	
Reproducibility:	Rarely	Fixed Version:	2023-06-13
Test Type:	Monkey Test	Root cause:	libdolphin报空指针，高通提供方案，禁用libdolphin
Description			
【执行命令】 monkey --pkg-blacklist-file /sdcard/blacklist.txt --ignore-crashes --ignore-timeouts --ignore-security-exceptions --monitor-native-crashes --ignore-native-crashes -s 20 --throttle 500 --pct-touch 30 --pct-motion 15 --pct-nav 15 --pct-majornav 15 --pct-syskeys 5 --pct-appswitch 10 --pct-anyevent 10 --ignore-crashes -v -v -v 2500000 2>/sdcard/error.txt 1>/sdcard/info.txt			
【执行结果】 monkey自动停止，进入屏保			
Subtasks:			
Bug # 118328: 【AF】 【EVT3】 【Monkey】 Device#3(4+27个APK（无豌豆荚）-客户测试命令...			CLOSED
Bug # 118042: IOVDEV-28972 【内部】 【VC1】 【0095】 【台架】 【EE环境】 【副屏】 偶发一次日志...			CLOSED
Related issues:			
Related to Figure - Bug # 118598: 【AF】 【EVT3】 【Monkey】 107版本...			CLOSED 2023-06-07

History

#1 - 2023-05-29 10:12 - CD TEST-方永红

ftp://cdiot@192.168.87.46/Pre_figure/Test_Log/118327%26328/Device%232

#2 - 2023-05-29 11:18 - CD SYSTEM-夏旭

1. 确认system_server重启，从下面的log可能明确system_server重启

```
./tlog_iiov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0020_0528221309/prop.txt:1092:[sys.system_server.start_count]:  
[1]  
./tlog_iiov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0021_0528223643/prop.txt:1095:[sys.system_server.start_count]:  
[2]
```

2. 依次排查下面目录里面的log确认system_server为什么会重启

```
./tlog_iiov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0020_0528221309/  
此目录下的log 显示 system_server log pid 未更改，说明system_server 未重启  
./tlog_iiov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0021_0528223643/  
在此目录下发现sytem_server pid 在下面两个log文件中已经发生了变化，  
./logcat/000372_persist_04138_230528_223619.log:1169:05-28 22:36:22.286 1552 1678 E system_server: No package ID ff found for ID 0xffffffff.  
./logcat/000372_persist_04139_230528_223624.log:1075:05-28 22:36:25.594 20715 20715 W system_server: Failed to upload odrefresh metrics: metrics file  
'/data/misc/odrefresh/odrefresh-metrics.txt' could not be opened: No such file or directory  
pid从1552变成了20715
```

3. 依次排查下面log文件中，看什么原因造成system_server 重启

- 3.1 从下面的日志文件发现surfaceflinger crash*

```
tlog_i ov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0021_0528223643/logcat/000372_persist_04138_230528_223619.log
05-28 22:36:23.102 810 3736 D audio_hw_hfp: hfp_set_parameters: enter
05-28 22:36:23.104 20578 20578 F DEBUG : * *
05-28 22:36:23.104 20578 20578 F DEBUG : Build fingerprint: 'HiPhi/PassengerDisplay_CN/PassengerDisplay:12/SKQ1.220201.001/3642:user/test-keys'
05-28 22:36:23.104 20578 20578 F DEBUG : Revision: '0'
05-28 22:36:23.104 20578 20578 F DEBUG : ABI: 'arm64'
05-28 22:36:23.104 20578 20578 F DEBUG : Timestamp: 2023-05-28 22:36:22.726604562+0800
05-28 22:36:23.104 20578 20578 F DEBUG : Process uptime: 35610s
05-28 22:36:23.104 20578 20578 F DEBUG : Cmdline: /system/bin/surfaceflinger
05-28 22:36:23.104 20578 20578 F DEBUG : pid: 912, tid: 1705, name: Binder:912_3 >>> /system/bin/surfaceflinger<<<
05-28 22:36:23.104 20578 20578 F DEBUG : uid: 1000
05-28 22:36:23.105 20578 20578 F DEBUG : signal 6 (SIGABRT), code 1 (SI_QUEUE), fault addr -----
05-28 22:36:23.105 20578 20578 F DEBUG : Abort message: 'Scudo ERROR: invalid chunk state when deallocating address 0x200007368ff4170'
05-28 22:36:23.105 20578 20578 F DEBUG : '
05-28 22:36:23.106 20578 20578 F DEBUG : x0 0000000000000000 x1 000000000000006a9 x2 0000000000000006 x3 0000007307c62ba0
05-28 22:36:23.106 20578 20578 F DEBUG : x4 0000000000808080 x5 0000000000808080 x6 0000000000808080 x7 8080800000000000
05-28 22:36:23.106 20578 20578 F DEBUG : x8 00000000000000f0 x9 c3ffa1715599bbc2 x10 0000000000000000 x11 ffffff80fffffdff
05-28 22:36:23.106 20578 20578 F DEBUG : x12 0000000000000001 x13 0000009f3b15a8a7 x14 0022550feb8ff800 x15 0000000000000070
05-28 22:36:23.106 20578 20578 F DEBUG : x16 000000762c8d9060 x17 000000762c8b5560 x18 00000073049f0000 x19 00000000000000390
05-28 22:36:23.106 20578 20578 F DEBUG : x20 000000000000006a9 x21 00000000ffffff x22 0000000000000000 x23 b400007378cc0930
05-28 22:36:23.106 20578 20578 F DEBUG : x24 0000007316830000 x25 0000007307c62e38 x26 0000007307c63270 x27 0000007307c631d8
05-28 22:36:23.106 20578 20578 F DEBUG : x28 0000007307c62ff0 x29 0000007307c62c20
05-28 22:36:23.106 20578 20578 F DEBUG : lr 000000762c86595c sp 0000007307c62b80 pc 000000762c865988 pst 0000000000001000
```

3.1.1 服务启动脚本中指定，当surfaceflinger crash 时，会导致zygotet重启，zygotet重启必定导致system_server重启，所以system_server pid会变

```
service surfaceflinger /system/bin/surfaceflinger
    class core animation
    user system
    group graphics drmrpc readproc
    capabilities SYS_NICE
    onrestart restart zygote -----> 会导致zygotet重启
    task_profiles HighPerformance
    socket pdx/system/vr/display/client stream 0666 system graphics u:object_r:pdx_display_client_endpoint_socket:s0
    socket pdx/system/vr/display/manager stream 0666 system graphics u:object_r:pdx_display_manager_endpoint_socket:s0
    socket pdx/system/vr/display/vsync stream 0666 system graphics u:object_r:pdx_display_vsync_endpoint_socket:s0
```

3.2 从下面的log能看到zygote重启

```
tlog_i ov0201017500006162025411a2212290000002271_000372_0529095401/FSEM_000372_0021_0528223643/logcat/000372_persist_04139_230528_223624.log
05-28 22:36:24.316 20599 20599 I zygote : option0=-Xzygote
05-28 22:36:24.316 20599 20599 I zygote : option1=-Xdeny-art-apex-data-files
05-28 22:36:24.316 20599 20599 I zygote : option2=exit
05-28 22:36:24.316 20599 20599 I zygote : option3=-vfprintf
05-28 22:36:24.316 20599 20599 I zygote : option4=-sensitiveThread
05-28 22:36:24.316 20599 20599 I zygote : option5=-verbose:gc
05-28 22:36:24.316 20599 20599 I zygote : option6=-XX:PerfettoHprof=true
```

4. 从上面的两个log文件中未发现system_server异常信息

5. 请surfaceflinger 的owner分析一下，谢谢

#3 - 2023-05-29 11:22 - CD SYSTEM-夏旭
- Category changed from SYSTEM to CD-FW
- Assignee changed from CD SYSTEM-夏旭 to CD FW 曹覃刚

Hi 覃刚:
请看一下surfaceflinger crash的问题，谢谢

#4 - 2023-05-29 14:11 - CD FW 曹覃刚
- Status changed from New to ASSIGNED

■ 我的分析
surfaceflinger报错，导致system_server重启，高通的libdolphin.so中报出来的问题

05-28 22:36:23.104 20578 20578 F DEBUG : *
05-28 22:36:23.104 20578 20578 F DEBUG : Build fingerprint: 'HiPhi/PassengerDisplay_CN/PassengerDisplay:12/SKQ1.220201.001/3642:user/test-keys'
05-28 22:36:23.104 20578 20578 F DEBUG : Revision: '0'
05-28 22:36:23.104 20578 20578 F DEBUG : ABI: 'arm64'
05-28 22:36:23.104 20578 20578 F DEBUG : Timestamp: 2023-05-28 22:36:22.726604562+0800
05-28 22:36:23.104 20578 20578 F DEBUG : Process uptime: 35610s
05-28 22:36:23.104 20578 20578 F DEBUG : Cmdline: /system/bin/surfaceflinger
05-28 22:36:23.104 20578 20578 F DEBUG : pid: 912, tid: 1705, name: Binder:912_3 >>> /system/bin/surfaceflinger <<<
05-28 22:36:23.104 20578 20578 F DEBUG : uid: 1000
05-28 22:36:23.105 20578 20578 F DEBUG : signal 6 (SIGABRT), code 1 (SI_QUEUE), fault addr -----
05-28 22:36:23.105 20578 20578 F DEBUG : Abort message: 'Scudo ERROR: invalid chunk state when deallocating address 0x200007368ff4170'
05-28 22:36:23.105 20578 20578 F DEBUG : '
05-28 22:36:23.106 20578 20578 F DEBUG : x0 0000000000000000 x1 000000000000006a9 x2 0000000000000006 x3 0000007307c62ba0
05-28 22:36:23.106 20578 20578 F DEBUG : x4 0000000000808080 x5 0000000000808080 x6 0000000000808080 x7 8080800000000000
05-28 22:36:23.106 20578 20578 F DEBUG : x8 00000000000000f0 x9 c3ffa1715599bbc2 x10 0000000000000000 x11 fffff80ffffffdf
05-28 22:36:23.106 20578 20578 F DEBUG : x12 0000000000000001 x13 0000009f3b15a8a7 x14 0022550feb8ff800 x15 0000000000000070
05-28 22:36:23.106 20578 20578 F DEBUG : x16 000000762c8d9060 x17 000000762c8b5560 x18 00000073049f0000 x19 0000000000000390
05-28 22:36:23.106 20578 20578 F DEBUG : x20 000000000000006a9 x21 00000000fffffffd x22 0000000000000000 x23 b400007378cc0930
05-28 22:36:23.106 20578 20578 F DEBUG : x24 0000007316830000 x25 0000007307c62e38 x26 0000007307c63270 x27 0000007307c631d8
05-28 22:36:23.106 20578 20578 F DEBUG : x28 0000007307c62ff0 x29 0000007307c62c20
05-28 22:36:23.106 20578 20578 F DEBUG : lr 000000762c86595c sp 0000007307c62b80 pc 000000762c865988 pst 0000000000001000
05-28 22:36:23.106 20578 20578 F DEBUG : backtrace:
05-28 22:36:23.106 20578 20578 F DEBUG : #00 pc 000000000051988 /apex/com.android.runtime/lib64/bionic/libc.so (abort+168) (BuildId: 369edc656806aeaf384cbeb8f7a347af)
05-28 22:36:23.106 20578 20578 F DEBUG : #01 pc 00000000004165c /apex/com.android.runtime/lib64/bionic/libc.so (scudo::die()+12) (BuildId: 369edc656806aeaf384cbeb8f7a347af)
05-28 22:36:23.106 20578 20578 F DEBUG : #02 pc 000000000041d74 /apex/com.android.runtime/lib64/bionic/libc.so (scudo::ScopedErrorReport::~ScopedErrorReport()+36) (BuildId: 369edc656806aeaf384cbeb8f7a347af)
05-28 22:36:23.106 20578 20578 F DEBUG : #03 pc 000000000041fc8 /apex/com.android.runtime/lib64/bionic/libc.so (scudo::reportInvalidChunkState(scudo::AllocatorAction, void)+80) (BuildId: 369edc656806aeaf384cbeb8f7a347af)
05-28 22:36:23.106 20578 20578 F DEBUG : #04 pc 000000000043664 /apex/com.android.runtime/lib64/bionic/libc.so (scudo::Allocator<scudo::AndroidConfig, &(scudo::malloc_posinit)>::deallocate(void*, scudo::Chunk::Origin, unsigned long, unsigned long)+316) (BuildId: 369edc656806aeaf384cbeb8f7a347af)

05-28 22:36:23.106 20578 20578 F DEBUG : #05 pc 000000000000aaf4 /system_ext/lib64/libdolphin.so (android::updateLayerListLocked(std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char> > const&, bool) (.cfi)+372) (BuildId: a2bd01334f3682675a19bcb37cfc7c99)

05-28 22:36:23.106 20578 20578 F DEBUG : #06 pc 000000000000b3e4 /system_ext/lib64/libdolphin.so (dolphinTrackBufferIncrement.cfi+436) (BuildId: a2bd01334f3682675a19bcb37cfc7c99)

05-28 22:36:23.106 20578 20578 F DEBUG : #07 pc 00000000001c05b8 /system/bin/surfaceflinger (ZNSt3__110__function6__funcIN7android14SurfaceFlinger19setTransactionStateERKNS2_17FrameTimelineInfoERKNS2_6VectorINS2_13ComposerStateEEERKNS7_INS2_12DisplayStateEEEjRKNS2_2spINS2_7IBinderEEERKNS2_19InputWindowCommandsElbRKNS2_14client_cache_tEbRKNS_6vectorINS2_17ListenerCallbacksENS_9allocatorISS_EEEEmE4\$_65NST_ISY_EEFvRKNS2_13layer_state_tEEEcIES12_\$8fd14edc326db0e9be510f46fb0c4dd5+688) (BuildId: 8958ca1cdc030c3e3967c8ecef82dddb)

05-28 22:36:23.106 20578 20578 F DEBUG : #08 pc 00000000001a0ce0 /system/bin/surfaceflinger (android::SurfaceFlinger::setTransactionState(android::FrameTimelineInfo const&, android::Vector<android::ComposerState> const&, android::Vector<android::DisplayState> const&, unsigned int, android::sp<android::IBinder> const&, android::InputWindowCommands const&, long, bool, android::client_cache_t const&, bool, std::__1::vector<android::ListenerCallbacks, std::__1::allocator<android::ListenerCallbacks> > const&, unsigned long)+1724) (BuildId: 8958ca1cdc030c3e3967c8ecef82dddb)

05-28 22:36:23.106 20578 20578 F DEBUG : #09 pc 00000000000c44d8 /system/lib64/libgui.so (android::BnSurfaceComposer::onTransact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+13700) (BuildId: 448b4199bacacf9417cd93df4f7390f5)

05-28 22:36:23.106 20578 20578 F DEBUG : #10 pc 00000000001aa468 /system/bin/surfaceflinger (android::SurfaceFlinger::onTransact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+852) (BuildId: 8958ca1cdc030c3e3967c8ecef82dddb)

05-28 22:36:23.106 20578 20578 F DEBUG : #11 pc 0000000000049880 /system/lib64/libbinder.so (android::BBinder::transact(unsigned int, android::Parcel const&, android::Parcel*, unsigned int)+252) (BuildId: 07bc4d4147790b1e0ac44f7625168f4a)

05-28 22:36:23.106 20578 20578 F DEBUG : #12 pc 00000000000538cc /system/lib64/libbinder.so (android::IPCThreadState::executeCommand(int)+1048) (BuildId: 07bc4d4147790b1e0ac44f7625168f4a)

05-28 22:36:23.106 20578 20578 F DEBUG : #13 pc 00000000000533e8 /system/lib64/libbinder.so (android::IPCThreadState::getAndExecuteCommand()+164) (BuildId: 07bc4d4147790b1e0ac44f7625168f4a)

05-28 22:36:23.106 20578 20578 F DEBUG : #14 pc 0000000000053c98 /system/lib64/libbinder.so (android::IPCThreadState::joinThreadPool(bool)+72) (BuildId: 07bc4d4147790b1e0ac44f7625168f4a)

05-28 22:36:23.106 20578 20578 F DEBUG : #15 pc 0000000000080484 /system/lib64/libbinder.so (android::PoolThread::threadLoop()+28) (BuildId: 07bc4d4147790b1e0ac44f7625168f4a)

05-28 22:36:23.106 20578 20578 F DEBUG : #16 pc 00000000001358c /system/lib64/libutils.so (android::Thread::_threadLoop(void*)+264) (BuildId: b940f836006ead456c936e97e64d2d4)

05-28 22:36:23.106 20578 20578 F DEBUG : #17 pc 000000000012de8 /system/lib64/libutils.so (thread_data_t::trampoline(thread_data_t const*)+408) (BuildId: b940f836006ead456c936e97e64d2d4)

05-28 22:36:23.106 20578 20578 F DEBUG : #18 pc 0000000000b6a24 /apex/com.android.runtime/lib64/bionic/libc.so (_pthread_start(void*)+264) (BuildId: 369edc656806aef384cbeb8f7a347af)

05-28 22:36:23.106 20578 20578 F DEBUG : #19 pc 00000000000532bc /apex/com.android.runtime/lib64/bionic/libc.so (_start_thread+68) (BuildId: 369edc656806aef384cbeb8f7a347af)

■ 下一步计划
起个高通case跟踪

#5 - 2023-05-29 15:39 - CD FW 曹覃刚

■ 我的分析
已起高通case:06648370

■ 下一步计划
推进高通分析并更新反馈

#6 - 2023-05-30 14:54 - CD FW 曹覃刚

■ 我的分析

已起高通case:06648370
05/30,高通未回复

■ 下一步计划

推进高通分析并更新反馈

#7 - 2023-05-31 15:48 - CD FW 曹覃刚

■ 我的分析

已起高通case:06648370
05/30晚些时候,高通给了个patch,但是我们没有他们so库里的源码,无法直接用,让他们直接提供so库了
05/31,高通未回复,我们这边加了comment要他们的电话号码

■ 下一步计划

推进高通分析并更新反馈

#8 - 2023-06-01 18:08 - CD FW 曹覃刚

■ 我的分析

已起高通case:06648370
06/01,高通未回复,未提供电话

■ 下一步计划

推进高通分析并更新反馈

#9 - 2023-06-02 19:59 - CD FW 曹覃刚

■ 我的分析

已起高通case:06648370
06/02,高通未回复,我们再次更新了comment,看他们是否可以直接提供so库

■ 下一步计划

推进高通分析并更新反馈

#10 - 2023-06-05 17:50 - CD FW 曹覃刚

■ 我的分析

已起高通case:06648370
06/05,高通回复,说该SO库是他们另外的Performance组负责的,让我们重新起case给该组

■ 下一步计划

已新起case:06659299给Performance Team
推进高通分析新的case并更新反馈

#11 - 2023-06-06 21:51 - CD FW 曹覃刚

■ 我的分析

已起高通case:06659299
06/06, 高通问了下日志位置，我们已回复，之后高通就没回复了

■ 下一步计划
推进高通分析并更新反馈

#12 - 2023-06-07 19:19 - CD FW 曹覃刚

■ 我的分析
已起高通case:06659299
06/07, 高通回复如下

libdolphin.so is mainly used for game performance, you may just disable it by removing it from the device /system_ext/lib64, it won't affect surfaceflinger's process

让我们直接禁用libdolphin.so，说这个库主要用于游戏性能的

但是我们的设备是有游戏模式的，因此需要确认清楚对游戏的影响，已回复高通，待高通回复

■ 下一步计划
推进高通分析并更新反馈

#13 - 2023-06-08 22:08 - CD FW 曹覃刚

■ 我的分析
已起高通case:06659299
06/07, 高通回复如下
禁用libdolphin.so之后会不会对我们的游戏模式产生影响
让我们直接禁用后测试试试

■ 下一步计划
禁用libdolphin.so后测试看看

#14 - 2023-06-10 17:53 - CD FW 曹覃刚

■ 我的分析
已起高通case:06659299
禁用libdolphin.so后测试看看，已编译版本，待测试确认影响
06/10, 继续追问了高通禁用之后的影响

■ 下一步计划
1.禁用libdolphin.so后测试看看，版本已出，依赖测试同学测试看看，主要是游戏模式相关功能
/Pre_figure/VerifyBuild/Pre_figure_turbox-c2130c-la1.1-qssi12-dev/20230610/202306101417-4300
2.继续与高通确认 禁用libdolphin.so 的影响

#15 - 2023-06-12 17:16 - IoT scm

Gerrit Merge Information :

ID	Project	Branch	Uploader
206151	general/platform/frameworks/native	Pre_figure_turbox-c2130c-la1.1-qssi12	caoqg0702@thundersoft.com

		-dev	
FW:SystemOptimization: Fix the pointer problem caused by libdolphinTC-RID: 1201-0205101IssueID: TS-R-BUG-118327Change-Id: I580d391738b3be1b64ebe4ec3390264ae60b583b			

#16 - 2023-06-12 17:21 - CD FW 曹覃刚

- Status changed from ASSIGNED to RESOLVED
- Assignee changed from CD FW 曹覃刚 to CD TEST-方永红
- Resolution changed from -- to FIXED
- Degrated changed from -- to No
- Fixed Version set to 2023-06-13
- Root cause set to libdolphin报空指针，高通提供方案，禁用libdolphin

■ 我的分析

已起高通case:06659299

1.06/12, 高通回复如下, 通常情况下, libdolphin是动态加载在sf, 如果lib加载失败, 或者平台不支持, 没有副作用, 就像以前一样按照正常的方式进行。

"normally, libdolphin is dynamically loaded in sf

frameworks/native/services/surfaceflinger/SurfaceFlinger.cpp

509 mDolphinHandle = dlopen("libdolphin.so", RTLD_NOW);

If lib loaded failed, or platform doesn't support, no side effect, just follow the normal way as like before."

2.VB测试验证游戏模式功能正常

3.客户意见:没明显副作用先入库

■ 下一步计划

合入禁用libdolphin.so的提交, 待正式版本验证

<https://dev.thundercomm.com/gerriit/c/general/platform/frameworks/native/+206151>

#17 - 2023-07-25 16:23 - CD TEST-方永红

- Status changed from RESOLVED to VERIFIED
- 7.25
- 该问题给出了明确的修复方案, 如有新的monkey停止再另起票
- 先关闭此问题

#18 - 2023-07-25 16:23 - CD TEST-方永红

- Status changed from VERIFIED to CLOSED