


Comments (4)DependenciesDuplicates (0)Blocking (0)Resources (0)

Infeasible

BugP3

+ Add Hotlist

 STATUS UPDATE No update yet.

Edit

 DESCRIPTION jm...@gmail.com created issue #1

Jul 9, 2019 04:47PM



```
pid: 1504, tid: 3986, name: Binder:1504_1C >>> system_server <<<
signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 0x3fe0000000000000
x0 00000075b9504e48 x1 00000075b9504f48 x2 000000001452be30 x3 00000075b9504e20
x4 00000075b9505228 x5 00000075b95051c0 x6 0000000000000010 x7 00000075da1d3480
x8 3fe0000000000000 x9 0000000000000004 x10 0000000000000005 x11 0000000076776d84
x12 00000075b9505110 x13 00000075de6064c8 x14 00000075f6056be4 x15 0000000000000000
x16 0000007679234110 x17 0000007677218b14 x18 0000000000000008 x19 00000075b9504e20
x20 00000075b9504f48 x21 00000075b9504e48 x22 0000000000000001 x23 00000075f60a9a38
x24 00000000776f112f x25 00000075b9505588 x26 0000000000000000 x27 0000000000000043
x28 00000075b9505588 x29 00000075b9504e10
sp 00000075b9504de0 lr 00000075f5f5e710 pc 00000075f5f5d230

backtrace:
#00 pc 000000000045e230 /system/lib64/libart.so (art::(anonymous
namespace)::ArgArray::BuildArgArrayFromVarArgs(art::ScopedObjectAccessAlreadyRunnable const&, art::ObjPtr<art::mirror::Object>,
std::__va_list)+280)
#01 pc 000000000045f70c /system/lib64/libart.so
(art::InvokeVirtualOrInterfaceWithVarArgs(art::ScopedObjectAccessAlreadyRunnable const&, _jobject*, _jmethodID*,
std::__va_list)+416)
#02 pc 0000000000334f64 /system/lib64/libart.so (art::JNI::CallBooleanMethodV(_JNIEnv*, _jobject*, _jmethodID*,
std::__va_list)+656)
#03 pc 0000000000c6be8 /system/lib64/libandroid_runtime.so (_JNIEnv::CallBooleanMethod(_jobject*, _jmethodID*, ...)+116)
#04 pc 0000000000137b2c /system/lib64/libandroid_runtime.so (JavaBBinder::onTransact(unsigned int, android::Parcel const&,
android::Parcel*, unsigned int)+156)
#05 pc 00000000004fb1c /system/lib64/libbinder.so (android::BBinder::transact(unsigned int, android::Parcel const&,
android::Parcel*, unsigned int)+136)
#06 pc 00000000005d1cc /system/lib64/libbinder.so (android::IPCThreadState::executeCommand(int)+520)
#07 pc 00000000005cf10 /system/lib64/libbinder.so (android::IPCThreadState::getAndExecuteCommand()+156)
#08 pc 00000000005d600 /system/lib64/libbinder.so (android::IPCThreadState::joinThreadPool(bool)+108)
#09 pc 00000000007fae8 /system/lib64/libbinder.so (android::PoolThread::threadLoop()+24)
#10 pc 0000000000100dc /system/lib64/libutils.so (android::Thread::_threadLoop(void*)+284)
#11 pc 0000000000b673c /system/lib64/libandroid_runtime.so (android::AndroidRuntime::javaThreadShell(void*)+140)
#12 pc 000000000090114 /system/lib64/libc.so (__pthread_start(void*)+36)
#13 pc 0000000000239a8 /system/lib64/libc.so (__start_thread+68)
```

COMMENTS

All comments ▾

↓ Oldest first



ku...@google.com <ku...@google.com>

Assigned to ku...@google.com.

Jul 11, 2019 06:49PM



jm...@gmail.com <jm...@gmail.com>

Jul 11, 2019 07:05PM

 deleted

 Restricted

0 B 



ku...@google.com <ku...@google.com> #2

Jul 11, 2019 08:27PM



Can you provide the below requested information to better understand the issue:


Android build
Which Android build are you using? (e.g. OPM1.171019.011)

Device used
Which device did you use to reproduce this issue?

Issue details
Please explain the issue in detail.

Steps to reproduce
What steps are needed to reproduce this issue?
Please explain the reproduction steps in detail.

Reporter

 jm...@gmail.com

Type

Bug

Priority

P3

Severity

S3

Status


Won't fix (Infeasible)

Access

Default access

View


Assignee

 ku...@google.com


Verifier

--

Collaborators

 ^

CC

 jm...@gmail.com ^

AOSP ID

--

ReportedBy

--

Found In

--


Targeted To

--

Verified In

--

In Prod





jm...@gmail.com <jm...@gmail.com> [#3](#)

Jul 16, 2019 05:40PM ⋮

The android build is Android P.
used xiaomi device.
the online bug report got this issue.
and only have the backtrace and log.
didn't know the reproduction steps!

BR!



ku...@google.com <ku...@google.com> [#4](#)

Sep 6, 2019 05:35PM ⋮

Status: Won't Fix (Infeasible)

Thank you for reporting this issue. Unfortunately, this issue is not reproducible with our version of the AOSP source code for Pixel devices. Please file a bug with the respective OEM, as they are using their own customized version of the AOSP source code. This bug will be marked as closed.