

← ↻ ☆ JNI DETECTED ERROR IN APPLICATION: field operation on NULL object: 0x0

+1

Hotlists (2)

Mark as Duplicate

🔔

⋮

Comments (5)

Dependencies

Duplicates (0)

Blocking (0)

Resources (2)

Fixed

Bug

P3

+ Add Hotlist

[AOSP] assigned

👤 STATUS UPDATE

No update yet.

Edit

📄 DESCRIPTION

pa...@mercurydevelopment.com created issue #1

Sep 10, 2018 09:49PM

⋮

08-14 16:14:58.462 18426-18437/com.eapp.manet A/app.mane: java_vm_ext.cc:542] JNI DETECTED ERROR IN APPLICATION: field operation on NULL object: 0x0
java_vm_ext.cc:542] in call to GetIntField
java_vm_ext.cc:542] from void sun.nio.ch.FileDispatcherImpl.preClose0(java.io.FileDescriptor)
java_vm_ext.cc:542] "FinalizerDaemon" daemon prio=5 tid=8 Runnable
java_vm_ext.cc:542] | group="system" sCount=0 dsCount=0 flags=0 obj=0x133003d0 self=0x7b003d3c00
java_vm_ext.cc:542] | sysTid=18437 nice=4 cgrp=default sched=0/0 handle=0x7aeb5304f0
java_vm_ext.cc:542] | state=R schedstat=(59165366 28905416 77) utm=5 stm=0 core=3 HZ=100
java_vm_ext.cc:542] | stack=0x7aeb42d000-0x7aeb42f000 stackSize=1041KB
java_vm_ext.cc:542] | held mutexes= "mutator lock"(shared held)
java_vm_ext.cc:542] native: #00 pc 00000000003c8d04 /system/lib64/libart.so
(art::DumpNativeStack(std::__1::basic_ostream<char, std::__1::char_traits<char>>&, int, BacktraceMap*, char const*, art::ArtMethod*, void*, bool)+220)
java_vm_ext.cc:542] native: #01 pc 0000000000498b90 /system/lib64/libart.so
(art::Thread::DumpStack(std::__1::basic_ostream<char, std::__1::char_traits<char>>&, bool, BacktraceMap*, bool) const+352)
java_vm_ext.cc:542] native: #02 pc 00000000002e9bb8 /system/lib64/libart.so (art::JavaVMExt::JniAbort(char const*, char const*)+972)

Full crash in previous ticket <https://issuetracker.google.com/issues/112575287>

I created:
- minimal app for reproduce the bug (bugtest folder)
- bugreport (bugreport-marlin-PPR1.180610.009-2018-09-10-15-38-49.zip and bugreport-marlin-PPR1.180610.009-2018-09-10-15-38-49-dumpstate_log-20556.txt).
and share it with android-bugreport@google.com
<https://drive.google.com/open?id=1AgXlo3N-ZCcUuPbbmJdvOhwglKt2kgYl>

Precondition:
Device: Pixel XL
OS version: 9
Build number: PPR1.180610.009
WiFi with internet access.
Disable mobile data(GPRS....)

Steps for reproduce:
1) compile and install app
2) enable wifi via notification drawer
3) wait 2-5 sec
4) disable wifi via notification drawer
5) wait 10-20 sec
6) if app not crashed, then retry 2-5

Current output:
Application closed with unclear exception.

Expected output:
Application not closed. Or any normal exception, that can be fixed.

Frequency: 80-90%

You can uncomment
connectivityManager.bindProcessToNetwork(null);
in NetworkSwitcher.java, and crash will be gone. But it is very strange behavior, and absolutely unclear stacktrace.

Reporter

pa...@mercurydevelopment.c...

Type Bug

Priority P3

Severity S3

Status

Fixed

Access Default access View

Assignee

ku...@google.com

Verifier --

Collaborators

👤

CC

🔔

pa...@mercurydevelopment.com

AOSP ID --

ReportedBy --

Found In --

Targeted To --

Verified In --

In Prod

✓ Mentioned issues (1)

Hide all

P3

JNI DETECTED ERROR IN APPLICATION: field operation on NULL object: 0x0

"<https://issuetracker.google.com/112575287>" pa...@ #1

COMMENTS

All comments

↓ Oldest first

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

Sep 11, 2018 07:52PM

⋮

Assigned to ku...@google.com.

Thank you for reporting this issue. We have shared this with our product and engineering team and will update this issue with more information as it becomes available.



vi...@google.com <vi...@google.com> [#3](#)

Mar 29, 2019 08:18PM ⋮

Marked as fixed.

The issue has been fixed and it will become available in a future Android release.



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

Apr 29, 2019 07:43PM ⋮

Status: Assigned (reopened)

Please ignore [comment #3](#).



vi...@google.com <vi...@google.com> [#5](#)

Apr 29, 2019 09:27PM ⋮

Thanks for reporting this bug which was present in Android for several years.

Prior to Android P it was harmless, but an unrelated change in P causes it to crash in native code. The bug is fixed in the public Q betas.

On Android P you can work around it by avoiding calls to `ConnectivityManager.bindProcessToNetwork()`



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

May 22, 2019 06:05PM

Marked as fixed.