CHECK(format->findString("mime", &mime)) in android_media_MediaCodec.cpp causes crashes

+1 | Hotlists (4) | Mark as Duplicate | 🔔 | ⋮

Comments (8) Dependencies Duplicates (0) Blocking (0) Resources (1)

Assigned | Bug | P2 | + Add Hotlist | [AOSP] assigned

👥 **STATUS UPDATE** No update yet. | Edit

📄 **DESCRIPTION** hz...@gmail.com created issue #1

There are quite a few crashes like this from an app with a large installation base:

```
Thread: frameworks/base/media/jni/android_media_MediaCodec.cpp:345 CHECK(format->findString("mime", &mime)) failed.

backtrace:
  #00  pc 0x0000000000051ba8  /apex/com.android.runtime/lib64/bionic/libc.so (abort+164)
  #01  pc 0x00000000006f365c  /apex/com.android.art/lib64/libart.so (art::Runtime::Abort(char const*)+596)
  #02  pc 0x0000000000016ea8  /apex/com.android.art/lib64/libbase.so (android::base::SetAborter(std::__1::function<void (char const*)>&&)::$_3::__invoke(char
  #03  pc 0x0000000000006e10  /system/lib64/liblog.so (__android_log_assert+308)
  #04  pc 0x0000000000048758  /system/lib64/libmedia_jni.so (android::JMediaCodec::configure(android::sp<android::AMessage> const&, android::sp<android::IGra
  #05  pc 0x000000000004d130  /system/lib64/libmedia_jni.so (android_media_MediaCodec_native_configure(_JNIEnv*, _jobject*, _jobjectArray*, _jobjectArray*, _
  #06  pc 0x000000000001d9908  /system/framework/arm64/boot-framework.oat (art_jni_trampoline+184)
  #07  pc 0x00000000004d8264  /system/framework/arm64/boot-framework.oat (android.media.MediaCodec.configure+1524)
  #08  pc 0x00000000004d8b88  /system/framework/arm64/boot-framework.oat (android.media.MediaCodec.configure+72)
  #09  pc 0x000000000020a2b0  /apex/com.android.art/lib64/libart.so (nterp_helper+4016)
```

I wonder if CHECK(format->findString("mime", &mime)) throws an uncaught exception. If it is true, would it be better to handle the exception or surface the exception instead of causing a crash?

✓ **Links (1)**

🔗 **Links (1)**

"For steps to capture a bug report, please refer: https://developer.android.com/studio/debug/bug-report#bugreportdevice"

COMMENTS

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

Assigned to vi...@google.com.

Thank you for reporting this issue. For us to further investigate this issue, please provide the following additional information:

What steps are needed to reproduce this issue? Frequency of occurrence?

Which Android build are you using? (e.g. TQ3A.230705.001.A1)

Which device did you use to reproduce this issue?

Can you confirm if this issue is reproducible on a Pixel/Nexus device?

Please provide a sample project or apk to reproduce the issue. Also mention the steps to be followed for reproducing the issue with the given sample project or apk.

Android bug report (to be captured after reproducing the issue)
For steps to capture a bug report, please refer: https://developer.android.com/studio/debug/bug-report#bugreportdevice

Alternate method
Navigate to "Developer options", ensure "USB debugging" is enabled, then enable "Bug report shortcut". Capture bug report by holding the power button and selecting the "Take bug report" o

Note: Please upload the bug report and screenshot to google drive and share the folder to android-bugreport@google.com, then share the link here.

hz...@gmail.com <hz...@gmail.com> #3

The crashes are reported by Google Play Console. They are relatively frequent and rare in the absolute sense because there are hundreds of thousands of installs. According to GPC, this one

I have not been able to reproduce it but I will try it and supply the requested information accordingly.
The attached screenshots show the 36 crashes in the past month.

📎 **Wed Sep 06 2023 15:20:55 GMT-0400 (Eastern Daylight Time).png**
265 KB View Download ⊘

📎 **Wed Sep 06 2023 15:21:04 GMT-0400 (Eastern Daylight Time).png**
393 KB   View   Download ⑦

◯ **vi...@google.com** <vi...@google.com> #4

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

◯ **hz...@gmail.com** <hz...@gmail.com> #5

I tried to reproduce it by using a dummy mime for MediaCodec configurations on a physical device and a Pixel emulator but failed to produce any crashes. I will add entries for this in the app

---

◯ **hz...@gmail.com** <hz...@gmail.com> #6

This happens to Android 12 too:

Thread frameworks/base/media/jni/android_media_MediaCodec.cpp:345 CHECK(format->findString("mime", &mime)) failed.

```
backtrace:
  #00  pc 0x0000000000078b9c  /apex/com.android.runtime/lib64/bionic/libc.so (abort+180)
  #01  pc 0x00000000006f331c  /apex/com.android.art/lib64/libart.so (art::Runtime::Abort(char const*)+596)
  #02  pc 0x0000000000016ea8  /apex/com.android.art/lib64/libbase.so (android::base::SetAborter(std::__1::function<void (char const*)>&&)::$_3::__invoke(c
  #03  pc 0x0000000000006e8c  /system/lib64/liblog.so (__android_log_assert+308)
  #04  pc 0x00000000000456cc  /system/lib64/libmedia_jni.so (android::JMediaCodec::configure(android::sp<android::AMessage> const&, android::sp<android::I
  #05  pc 0x000000000004a2a0  /system/lib64/libmedia_jni.so (android_media_MediaCodec_native_configure(_JNIEnv*, _jobject*, _jobjectArray*, _jobjectArray*
  #06  pc 0x0000000000392fd8  /data/misc/apexdata/com.android.art/dalvik-cache/arm64/boot.oat (art_jni_trampoline+184)
```

Now, the app searches the codec list and ensures the device has a codec supporting the requested MIME (e.g., video/avc) before trying to create it.

📎 **Sat Sep 09 2023 10:17:53 GMT-0400 (Eastern Daylight Time).png**
67 KB   View   Download

◯ **hz...@gmail.com** <hz...@gmail.com> #7

I have just seen a crash report from an Android 11 device:

```
backtrace:
  #00  pc 0x000000000008cd24  /apex/com.android.runtime/lib64/bionic/libc.so (abort+164)
  #01  pc 0x0000000000531cd4  /apex/com.android.art/lib64/libart.so (art::Runtime::Abort(char const*)+2340)
  #02  pc 0x000000000001394c  /system/lib64/libbase.so (android::base::SetAborter(std::__1::function<void (char const*)>&&)::$_3::__invoke(char const*)+76
  #03  pc 0x0000000000006d10  /system/lib64/liblog.so (__android_log_assert+336)
  #04  pc 0x0000000000003fc54  /system/lib64/libmedia_jni.so (android::JMediaCodec::configure(android::sp<android::AMessage> const&, android::sp<android::I
  #05  pc 0x0000000000043798  /system/lib64/libmedia_jni.so (android_media_MediaCodec_native_configure(_JNIEnv*, _jobject*, _jobjectArray*, _jobjectArray*
  #06  pc 0x0000000000222520  /system/framework/arm64/boot-framework.oat (art_jni_trampoline+272)
  #07  pc 0x000000000060f690  /system/framework/arm64/boot-framework.oat (android.media.MediaCodec.configure+1472)
  #08  pc 0x00000000006102f4  /system/framework/arm64/boot-framework.oat (android.media.MediaCodec.configure+68)
  #09  pc 0x0000000000132564  /apex/com.android.art/lib64/libart.so (art_quick_invoke_stub+548)
  #10  pc 0x0000000000196e94  /apex/com.android.art/lib64/libart.so (art::ArtMethod::Invoke(art::Thread*, unsigned int*, unsigned int, art::JValue*, char
  #11  pc 0x0000000000303384  /apex/com.android.art/lib64/libart.so (art::interpreter::ArtInterpreterToCompiledCodeBridge(art::Thread*, art::ArtMethod*, a
  #12  pc 0x00000000002fe49c  /apex/com.android.art/lib64/libart.so (bool art::interpreter::DoCall<false, false>(art::ArtMethod*, art::Thread*, art::Shado
  #13  pc 0x0000000000637dcc  /apex/com.android.art/lib64/libart.so (MterpInvokeVirtualQuick+708)
  #14  pc 0x0000000000130594  /apex/com.android.art/lib64/libart.so (mterp_op_invoke_virtual_quick+20)
```

📎 **Thu Sep 14 2023 07:30:53 GMT-0400 (Eastern Daylight Time).png**
69 KB   View   Download

◯ **hz...@gmail.com** <hz...@gmail.com> #8

I should have mentioned that I tried to reproduce this by using a random MIME not supported by Android (e.g., "dummy/debug") to instantiate MediaCodec. It throws an exception but does no
MediaCodec with it but the crash still occurs.