Android Public Tracker > Framework   271322682

cts case android.nativemedia.aaudio.AAudioTests#test_attributes_aaudio_usage_perfnone fail due to AOSP code

+1 1   Hotlists (4)   Mark as Duplicate

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

Assigned   Bug   P3   + Add Hotlist   [AOSP] assigned

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

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

PLEASE READ:

This form is only for reporting bugs found in the Android system while
developing Android applications. Use the Tools templates for issues with
the developer tools.

Please describe the problem in detail. Be sure to include:
- Steps to reproduce the problem (including sample code if appropriate).
run cts -m CtsNativeMediaAAudioTestCases
- What happened.
java.lang.AssertionError: cts/tests/tests/nativemedia/aaudio/jni/test_aaudio_attributes.cpp:71 Expected equality of these values: AAUDIO_OK Which is: 0 AAudioStreamBuilder_openStream(aau
com.android.gtestrunner.GtestRunner.run(GtestRunner.java:87) at org.junit.runners.Suite.runChild(Suite.java:128) at org.junit.runners.Suite.runChild(Suite.java:27) at org.junit.runners.ParentRunn
org.junit.runners.ParentRunner.run(ParentRunner.java:363) at org.junit.runner.JUnitCore.run(JUnitCore.java:137) at org.junit.runner.JUnitCore.run(JUnitCore.java:115) at androidx.test.internal.run
- What you think the correct behavior should be.
The AOSP code frameworks/av/media/libaaudio/src/core/AudioStreamBuilder.cpp AS follow:
191 aaudio_result_t AudioStreamBuilder::validate() const {
192 193   // Check for values that are ridiculously out of range to prevent math overflow exploits.194   // The service will do a better check.195   aaudio_result_t result = AAudioStreamParame
196   if (result != AAUDIO_OK) {
197   return result;
198   }
199 200   switch (mPerformanceMode) {
201   case AAUDIO_PERFORMANCE_MODE_NONE:
202   case AAUDIO_PERFORMANCE_MODE_POWER_SAVING:
203   case AAUDIO_PERFORMANCE_MODE_LOW_LATENCY:
204   break;
205   default:
206   ALOGE("illegal performanceMode = %d", mPerformanceMode);
207   return AAUDIO_ERROR_ILLEGAL_ARGUMENT;
208   // break;209   }

But the test case fill unsupport parameters:
cts/tests/tests/nativemedia/aaudio/jni/test_aaudio_stream_builder.cpp
359 INSTANTIATE_TEST_CASE_P(PM, AAudioStreamBuilderPerfModeTest,
360   ::testing::Values(
361   // Reasonable values
362   AAUDIO_PERFORMANCE_MODE_NONE,
363   AAUDIO_PERFORMANCE_MODE_POWER_SAVING,
364   AAUDIO_PERFORMANCE_MODE_LOW_LATENCY,
365   // Odd values
366   AAUDIO_UNSPECIFIED - 1, AAUDIO_UNSPECIFIED, 100, 1000000, 10000000),
367   &AAudioStreamBuilderPerfModeTest::getTestName);

Don't forget to mention which version of Android you're using, and/or which
device the problem appears on (model and Android version).
Android 10 AOSP

attach is the cts logcat

📎 **deleted**
0 B ⓘ

📎 **deleted**
0 B ⓘ

✓ Links (1)

🔗 Links (1)

" ...he test failed when creating usage as VOICE_COMMUNICATION. From the error log(see related code for error logging 🔗here), it looks like the device only support voice communication at DIRECT

COMMENTS

⚪ **ch...@gmail.com** <ch...@gmail.com> #2

android 12(AOSP) has the same problem

---

su...@google.com <su...@google.com> #3

Assigned to su...@google.com.

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

---

ph...@google.com <ph...@google.com> #4

Reassigned to ch...@gmail.com.

Thank you for this report. There are a number of interesting things in this log:

03-01 21:59:33.639 5654 5675 I AAudioTest: Feature android.hardware.audio.output: not supported
03-01 21:59:33.692 5654 5675 I AAudioTest: Feature android.hardware.microphone: not supported

Does that seem right? What kind of device is this?
It seems like almost all of the AAudio tests should be skipped but they are still being run.
That may be a bug in CTS.

Are you a manufacturing partner? If so then please use one of the Partner components or contact your Technical Account Manager (TAM).

---

ph...@google.com <ph...@google.com> #5

We will add checks in CTS to skip some tests if there is no speaker or microphone.

---

ch...@gmail.com <ch...@gmail.com> #6

Dear google,

Q: Does that seem right? What kind of device is this?
A: The log of this I level is not sure whether it is correct. The current device is a chip platform reference board.

i am not a manufacturing partner,
We downloaded the AOSP code, adapted it to our chip（car ivi soc）, and provided it to our customers for secondary development.

I'll check the android.hardware.audio.output.There is information being updated.

Thanks a lot.

---

ji...@google.com <ji...@google.com> #7

It looks like the test failed when creating usage as VOICE_COMMUNICATION. From the error log(see related code for error logging ⇔here), it looks like the device only support voice commu

changmm1123@, could you share your dumpsys of `media.audio_policy`? It will also be good to know when you are adapting AOSP code to your chip set, do you customize audio policy ma

```
03-01 22:00:55.541  6092  6113 I AudioStreamBuilder: rate    =        0, channels  = 0, format   = 0, sharing = SH, dir = OUTPUT
03-01 22:00:55.541  6092  6113 I AudioStreamBuilder: device =        0, sessionId = -1, perfMode = 10, callback: OFF with frames = 0
03-01 22:00:55.541  6092  6113 I AudioStreamBuilder: usage  =        2, contentType = 0, inputPreset = 0, allowedCapturePolicy = 0
03-01 22:00:55.541  6092  6113 D AudioStreamBuilder: build() MMAP not available because AAUDIO_PERFORMANCE_MODE_LOW_LATENCY not used.
03-01 22:00:55.541  6092  6113 D          : PlayerBase::PlayerBase()
03-01 22:00:55.541  6092  6113 D AudioStreamTrack: open(), request notificationFrames = 0, frameCount = 0
03-01 22:00:55.542  3084  3084 D AudioFlinger: Client defaulted notificationFrames to 966 for frameCount 1932
03-01 22:00:55.542  3084  3084 E AudioFlinger: createTrack_l() Bad parameter: sampleRate 16000 format 0x5, channelMask 0x00000003 for output 0xfd8dc5a371c
03-01 22:00:55.543  6092  6113 E IAudioFlinger: createTrack returned error -22
03-01 22:00:55.543  6092  6113 E AudioTrack: createTrack_l(35): AudioFlinger could not create track, status: -22 output 306783040
03-01 22:00:55.543  6092  6113 W AudioTrack: removeAudioDeviceCallback removing different callback!
03-01 22:00:55.543  6092  6113 E AudioStreamTrack: open(), initCheck() returned -22
```

---

ph...@google.com <ph...@google.com> #8

Also what version of Android are you using?