Android Public Tracker   36933380

SIGILL when throwing an exception under Android 1.6 and 2.1

+1 14   Hotlists (1)   Mark as Duplicate

Comments (23)   Dependencies   Duplicates (0)   Blocking (0)   Resources (1)

Obsolete   Bug   P3   + Add Hotlist   [AOSP] assigned

STATUS UPDATE No update yet. Edit

DESCRIPTION ke...@gmail.com created issue #1   Sep 20, 2011 04:25AM

I'm developing an application which runs under Android 1.6+ with a lot of C++ code. It triggers a SIGILL error each time I run it under Android 1.6 or 2.1 emulators (2.2+ is fine).

I'm using boost::future which needs exceptions so BOOST_NO_EXCEPTIONS must not be defined.

I'm using NDK r6b.

My jni/Application.mk content is :

```
APP_ABI := armeabi
APP_PLATFORM := android-4
APP_STL := gnustl_static
APP_CPPFLAGS += -fexceptions
APP_CPPFLAGS += -frtti
```

I finished to isolate the problem which seems to comes from exceptions, a simple exception thrown in any JNI method generates a SIGILL signal.

Aren't exceptions supported with latest NDK ?

My test case is :

```
---
LOGD("before throwing logic error");

try{
      throw std::logic_error("test");
}catch(const std::exception &e){
      LOGE("logic error catched %s", e.what());
}

LOGD("after throwing logic error");

try{
      LOGD("before throwing 1");
      throw 1;
}catch(...){
      LOGE("1 catched");
}

LOGD("after throwing 1");
---
```

Here are the logcats for each Android versions I tested :

* Android 1.6

```
D/visio-client-jni(  386): before throwing logic error
I/DEBUG   (   28): *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** ***
I/DEBUG   (   28): Build fingerprint: 'generic/sdk/generic/:1.6/Donut/20842:eng/test-keys'
I/DEBUG   (   28): pid: 386, tid: 394  >>> com.neolinks.visiodroid <<<
I/DEBUG   (   28): signal 4 (SIGILL), fault addr 80800000
I/DEBUG   (   28):  r0 80a333ab  r1 44f60a1c  r2 80a333ad  r3 80800000
I/DEBUG   (   28):  r4 0025d2a0  r5 80abea6c  r6 80a333ab  r7 44f60ce8
I/DEBUG   (   28):  r8 44f60d20  r9 42508f80  10 424ffc22  fp 00000176
I/DEBUG   (   28):  ip 80abeb1c  sp 44f60a10  lr 80a3a7c4  pc 80800000  cpsr a0000010
I/DEBUG   (   28):          #00  pc 00000000  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #01  pc 0023a7c0  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #02  pc 0023ac84  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #03  pc 0023b1d4  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
```

Reporter	○ ke...@gmail.com
Type	Bug
Priority	P3
Severity	S3
Status	Won't fix (Obsolete)
Access	Default access View
Assignee	○ an...@google.com
Verifier	--
Collaborators	
CC	an...@google.com di...@android.com en...@google.com ke...@gmail.com
AOSP ID	20176
ReportedBy	Developer
Found In	--
Targeted To	--
Verified In	--
In Prod	○

```
I/DEBUG   (   28):          #04  pc 000541a0  /system/lib/libdvm.so
I/DEBUG   (   28): stack:
I/DEBUG   (   28):     44f609d0  000000da
I/DEBUG   (   28):     44f609d4  000001b8
I/DEBUG   (   28):     44f609d8  0024ae90  [heap]
I/DEBUG   (   28):     44f609dc  afe0ea00  /system/lib/libc.so
I/DEBUG   (   28):     44f609e0  00002bcc
I/DEBUG   (   28):     44f609e4  afe0ed94  /system/lib/libc.so
I/DEBUG   (   28):     44f609e8  00002bcc
I/DEBUG   (   28):     44f609ec  00000003
I/DEBUG   (   28):     44f609f0  000000da
I/DEBUG   (   28):     44f609f4  000001b8
I/DEBUG   (   28):     44f609f8  0024adb0  [heap]
I/DEBUG   (   28):     44f609fc  afe0ea00  /system/lib/libc.so
I/DEBUG   (   28):     44f60a00  00002bcc
I/DEBUG   (   28):     44f60a04  afe0ed94  /system/lib/libc.so
I/DEBUG   (   28):     44f60a08  df002777
I/DEBUG   (   28):     44f60a0c  e3a070ad
I/DEBUG   (   28): #01 44f60a10  0024ae90  [heap]
I/DEBUG   (   28):     44f60a14  afe0ea00  /system/lib/libc.so
I/DEBUG   (   28):     44f60a18  00002bcc
I/DEBUG   (   28):     44f60a1c  afe0ed94  /system/lib/libc.so
I/DEBUG   (   28):     44f60a20  00002bcc
I/DEBUG   (   28):     44f60a24  44f60ca4
I/DEBUG   (   28):     44f60a28  0025d2a0  [heap]
I/DEBUG   (   28):     44f60a2c  44f60a48
I/DEBUG   (   28):     44f60a30  44f60ce8
I/DEBUG   (   28):     44f60a34  44f60d20
I/DEBUG   (   28):     44f60a38  42508f80
I/DEBUG   (   28):     44f60a3c  424ffc22  /data/dalvik-cache/data@app@com.neolinks.visiodroid.apk@classes.dex
I/DEBUG   (   28):     44f60a40  00000176
I/DEBUG   (   28):     44f60a44  80a3ac88  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so

* Android 2.1

D/visio-client-jni(  256): before throwing logic error
I/DEBUG   (   28): *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** ***
I/DEBUG   (   28): Build fingerprint: 'generic/sdk/generic/:2.1-update1/ECLAIR/35983:eng/test-keys'
I/DEBUG   (   28): pid: 256, tid: 263  >>> com.neolinks.visiodroid <<<
I/DEBUG   (   28): signal 4 (SIGILL), fault addr 80c00000
I/DEBUG   (   28):  r0 80e333ab  r1 46b589c4  r2 80e333ad  r3 80c00000
I/DEBUG   (   28):  r4 001164a0  r5 80ebea6c  r6 80e333ab  r7 46b58c90
I/DEBUG   (   28):  r8 46b58cc8  r9 42f0ff78  10 00000354  fp 42f0ff74
I/DEBUG   (   28):  ip 80ebeb1c  sp 46b589b8  lr 80e3a7c4  pc 80c00000  cpsr a0000010
I/VisioDroid(  256): getSupportedPreviewFpsRange not found, falling back to getSupportedPreviewFrameRates
I/DEBUG   (   28):          #00  pc 00000000  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #01  pc 0023a7c0  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #02  pc 0023ac84  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):          #03  pc 0023b1d4  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
I/DEBUG   (   28):
I/DEBUG   (   28): code around lr:
I/DEBUG   (   28): 80e3a7b4 0a000008 e1a00006 e28d100c ebfa7ae2
I/DEBUG   (   28): 80e3a7c4 e3500000 e1a09000 05840010 03a03009
I/DEBUG   (   28): 80e3a7d4 1a000007 ea000056 e59f3168 e59f2168
I/DEBUG   (   28):
I/DEBUG   (   28): stack:
I/DEBUG   (   28):     46b58978  00124490  [heap]
I/DEBUG   (   28):     46b5897c  afe0b39b  /system/lib/libc.so
I/DEBUG   (   28):     46b58980  0013f4c0  [heap]
I/DEBUG   (   28):     46b58984  0013f430  [heap]
I/DEBUG   (   28):     46b58988  00000000
I/DEBUG   (   28):     46b5898c  afe0f2c0  /system/lib/libc.so
I/DEBUG   (   28):     46b58990  0013f430  [heap]
I/DEBUG   (   28):     46b58994  00000000
I/DEBUG   (   28):     46b58998  00000000
I/DEBUG   (   28):     46b5899c  0011e588  [heap]
I/DEBUG   (   28):     46b589a0  00124490  [heap]
I/DEBUG   (   28):     46b589a4  0013f430  [heap]
I/DEBUG   (   28):     46b589a8  00000000
I/DEBUG   (   28):     46b589ac  46b58a0c
I/DEBUG   (   28):     46b589b0  df002777
I/DEBUG   (   28):     46b589b4  e3a070ad
I/DEBUG   (   28): #01 46b589b8  46b58a0c
I/DEBUG   (   28):     46b589bc  afe0f2c0  /system/lib/libc.so
I/DEBUG   (   28):     46b589c0  00000000
I/DEBUG   (   28):     46b589c4  000000c1
```

```
I/DEBUG   (   28):    46b589c8  0011e568  [heap]
I/DEBUG   (   28):    46b589cc  46b58c4c
I/DEBUG   (   28):    46b589d0  001164a0  [heap]
I/DEBUG   (   28):    46b589d4  46b589f0
I/DEBUG   (   28):    46b589d8  46b58c90
I/DEBUG   (   28):    46b589dc  46b58cc8
I/DEBUG   (   28):    46b589e0  42f0ff78
I/DEBUG   (   28):    46b589e4  00000354
I/DEBUG   (   28):    46b589e8  42f0ff74
I/DEBUG   (   28):    46b589ec  80e3ac88  /data/data/com.neolinks.visiodroid/lib/libvisio-client-jni.so
```

* Android 2.2 and 2.3

```
D/visio-client-jni(  360): before throwing logic error
E/visio-client-jni(  360): logic error catched test
D/visio-client-jni(  360): after throwing logic error
D/visio-client-jni(  360): before throwing 1
E/visio-client-jni(  360): 1 catched
D/visio-client-jni(  360): after throwing 1
```

✓ Links (1) Hide all

“Confirmed and testcase added https://android-review.googlesource.com/#/c/48451” an...@ #23

COMMENTS

All comments ▾ ↓ Oldest first

ti...@msn.com <ti...@msn.com> #2 Oct 13, 2011 09:46AM ⋮

I can reproduce this error under Android 1.5 on both emulator and device.

di...@android.com <di...@android.com> #3 Oct 20, 2011 12:38AM ⋮

I cannot reproduce this at all with NDK r6b and the following unit test (see below).
I really need a small reproducible test case, otherwise I won't be able to do anything about it :-(

```
-------------- cut here ----------------------------------------
/* This test is meant to check that C++ exceptions do not crash
 * when running on Eclair or older platform releases. It will
 * always succeed on later versions of the  platform!
 */
#include <new>
#include <exception>
#include <cstdio>

static int foo(void)
{
   try {
      ::printf("Hello ");
      throw std::exception();
   }
   catch (const std::exception &e) {
      ::printf(" World!\n");
   }
}

int main(int argc, char** argv)
{
   foo();
   return 0;
}
```

ke...@gmail.com <ke...@gmail.com> #4 Oct 20, 2011 12:43AM ⋮

OK I will create a small project with a JNI :)

Compiling a native console executable with exceptions is always working.

di...@android.com <di...@android.com> #5 Oct 20, 2011 12:47AM ⋮

Assigned to di...@android.com.

Hold on, I can reproduce it when building a shared library and calling it from JNI. It doesn't crash when I generate a stand-alone executable (as with the unit tests).

So it looks like an issue when gnustl_static is linked into a shared library instead of an executable. I'll look into this, but I'm not sure there is a simple solution yet (I suspect this is due to the lack of support for WEAK linking in the Android linker < 2.2)

ke...@gmail.com <ke...@gmail.com> #6 Oct 20, 2011 12:55AM

Thanks a lot, good luck :)

di...@android.com <di...@android.com> #7 Oct 25, 2011 03:40AM

For the record, I can reproduce the issue without JNI (which means I now have a unit test that runs from the adb shell directly, instead of having to build a complete .apk and launch it).

I don't have a solution though at the moment. I'm not even sure there will be one before the next NDK release.

ga...@gmail.com <ga...@gmail.com> #8 Nov 3, 2011 09:44PM

I have also met this problem, I had to revert back to ndk-r6.

di...@android.com <di...@android.com> #9 Nov 4, 2011 04:19AM

Are you saying that this works correctly with ndk-r6? If so that's an interesting lead.

ga...@gmail.com <ga...@gmail.com> #10 Nov 8, 2011 10:59PM

Yes, the same sources work for me with r6, but not if I compile with r6b.

di...@gmail.com <di...@gmail.com> #11 Nov 18, 2011 08:02AM

Does anyone know if this bug still exists for the newly released r7?

ti...@msn.com <ti...@msn.com> #12 Nov 18, 2011 09:05AM

This bug does still exist for ndk-r7.

ti...@msn.com <ti...@msn.com> #13 Nov 18, 2011 09:25AM

Actually, with ndk-r7, my app gets a SIGILL on creation if I call System.loadLibrary. So I'm not sure if it's the same bug.

ga...@gmail.com <ga...@gmail.com> #14 Jan 3, 2012 11:21PM

I can confirm, the bug exists in r7 too. The r6 is the last release, which does not has this problem.
Is there any progress with this?

92...@gmail.com <92...@gmail.com> #15 Jan 31, 2012 01:05PM

I had a same problem, but I think I fix...
it looks like a weak symbol problem, so I define func in my source file.

=== Start C++ unwind function redefine source ===

```
typedef long unsigned int *_Unwind_Ptr;

/* Stubbed out in libdl and defined in the dynamic linker.
 * Same semantics as __gnu_Unwind_Find_exidx().
 */
extern "C" _Unwind_Ptr dl_unwind_find_exidx(_Unwind_Ptr pc, int *pcount);
extern "C" _Unwind_Ptr __gnu_Unwind_Find_exidx(_Unwind_Ptr pc, int *pcount)
{
    return dl_unwind_find_exidx(pc, pcount);
}
```

```
===== end of file ======
```

and then force to binding symbol at JNI load

```
static void* g_func_ptr;
jint JNI_OnLoad(JavaVM *vm, void *reserved)
{
   // when i throw exception, linker maybe can't find __gnu_Unwind_Find_exidx(lazy binding issue??)
   // so I force to bind this symbol at shared object load time
   g_func_ptr = (void*)__gnu_Unwind_Find_exidx;
}
```

this works for me. I hope this also help you ;)

ke...@gmail.com <ke...@gmail.com> #16 Jan 31, 2012 07:11PM

Thanks a lot ! I will try your fix :)

ga...@gmail.com <ga...@gmail.com> #17 Feb 11, 2012 10:12PM

Thank you, this fixed my problem too.

za...@gmail.com <za...@gmail.com> #18 Feb 14, 2012 07:36AM

JNI_OnLoad should return the JNI version number:

```
jint JNI_OnLoad(JavaVM *vm, void *reserved)
{
   g_func_ptr = (void*)__gnu_Unwind_Find_exidx;
   return JNI_VERSION_1_6;
}
```

ke...@gmail.com <ke...@gmail.com> #19 Mar 19, 2012 08:01PM

Sorry for the late, it works under Android 1.6, thanks a lot 92soc...@gmail.com :)

David> Please could you fix the bug with workaround posted in comment #14 ?

I suppose missing variables/functions have to be added in some system static library.

st...@gmail.com <st...@gmail.com> #20 Jun 29, 2012 12:12AM

I have the same problem with ndk r8. The fix doesn't work for me. ndk-stack trace follows:

```
********** Crash dump: **********
Build fingerprint: 'generic/google_sdk/generic/:2.1/ERD79/22607:eng/test-keys'
pid: 369, tid: 382  >>> it.navionics.singleAppEurope <<<
signal 4 (SIGILL), fault addr 80b00000
Stack frame #00  pc 00000000  /data/data/it.navionics.singleAppEurope/lib/libgnustl_shared.so: Unable to locate routine information for address 0 in module obj/local/armeabi//libgnustl_shared.so
Stack frame #01  pc 000b209c  /data/data/it.navionics.singleAppEurope/lib/libgnustl_shared.so: Routine get_eit_entry in /i/ndk-andrewhsieh/src.1-with-cherrypicks//build/../gcc/gcc-4.4.3/libgcc/../gcc/config/arm/unwind-arm.c:603
Stack frame #02  pc 000b2560  /data/data/it.navionics.singleAppEurope/lib/libgnustl_shared.so: Routine __gnu_Unwind_RaiseException in /i/ndk-andrewhsieh/src.1-with-cherrypicks//build/../gcc/gcc-4.4.3/libgcc/../gcc/config/arm/unwind-arm.c:826
Stack frame #03  pc 000b2ab0  /data/data/it.navionics.singleAppEurope/lib/libgnustl_shared.so: Routine <unknown> in /i/ndk-andrewhsieh/src.1-with-cherrypicks//build/../gcc/gcc-4.4.3/libgcc/../gcc/config/arm/libunwind.S:334
```

tt...@gmail.com <tt...@gmail.com> #21 Jul 26, 2012 07:44PM

I have the same issue with r8b on android 2.1 (didn't test lower).
The fix posted by 92soc...@gmail.com worked for me.

bp...@gmail.com <bp...@gmail.com> #22 Nov 8, 2012 05:23AM

I also had the issue on 2.1-update1 using NDK r7. And the fix by 92soc...@gmail.com fixed exception handling for me too! :)

en...@google.com <en...@google.com> Dec 20, 2012 12:58PM

Reassigned to an...@google.com.

an...@google.com <an...@google.com> #23 Dec 20, 2012 05:08PM

Confirmed and testcase added https://android-review.googlesource.com/#/c/48451

en...@google.com <en...@google.com> Jun 27, 2015 06:18AM

Status: Won't Fix (Obsolete)