New issue

Jump to bottom

A Segmentation fault in cosprim.hh:49:13 #482

⊙ Open **seviezhou** opened this issue on Aug 25, 2020 · 7 comments

seviezhou commented on Aug 25, 2020

System info

Ubuntu x86_64, clang 6.0, faust (latest master c236d2)

Configure

cmake . -DCMAKE_CXX_FLAGS="-fsanitize=address -g" -DCMAKE_C_FLAGS="-fsanitize=address -g" -DCMAKE_EXE_LINKER_FLAGS="-fsanitize=address" -DINCLUDE_STATIC=on -DINCLUDE_HTTP=off -DINCLUDE_OSC=off

Command line

./build/bin/faust -lang ocpp -o /tmp/faust -e -lcc -exp10 -lb -rb -mem -sd @@

AddressSanitizer output

```
AddressSanitizer:DEADLYSIGNAL
=================================================================
==14194==ERROR: AddressSanitizer: SEGV on unknown address 0x000000000000 (pc 0x000000c0666a bp 0x7f2f710fc2f0 sp 0x7f2f710fc010 T1)
==14194==The signal is caused by a READ memory access.
==14194==Hint: address points to the zero page.
    #0 0xc06669 in isNum(CTree* const&, num&) /home/seviezhou/faust/compiler/signals/signals.hh:266:18
    #1 0xc5f65f in CosPrim::computeSigOutput(std::vector<CTree*, std::allocator<CTree*> > const&) /home/seviezhou/faust/compiler/extended/cosprim.hh:49:13
    #2 0x10d1c54 in simplification(CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:98:24
    #3 0x10cee26 in traced_simplification(CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:59:14
    #4 0x10cee26 in sigMap(CTree*, CTree* (*)(CTree*), CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:235
    #5 0x10cebcd in sigMap(CTree*, CTree* (*)(CTree*), CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:230:26
    #6 0x10cebcd in sigMap(CTree*, CTree* (*)(CTree*), CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:230:26
    #7 0x10cebcd in sigMap(CTree*, CTree* (*)(CTree*), CTree*) /home/seviezhou/faust/compiler/normalize/simplify.cpp:230:26
    #8 0xdf1f71 in numericBoxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1476:22
    #9 0xdf1f71 in boxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1436
    #10 0xe0fad9 in insideBoxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1639:19
    #11 0xdf1c0c in numericBoxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1488:18
    #12 0xdf1c0c in boxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1436
    #13 0xe0fad9 in insideBoxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1639:19
    #14 0xdf1c0c in numericBoxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1488:18
    #15 0xdf1c0c in boxSimplification(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:1436
    #16 0xdef2ac in evalprocess(CTree*) /home/seviezhou/faust/compiler/evaluate/eval.cpp:99:13
    #17 0xcbbc92 in evaluateBlockDiagram(CTree*, int&, int&) /home/seviezhou/faust/compiler/libcode.cpp:1146:20
    #18 0xcbbc92 in threadEvaluateBlockDiagram(void*) /home/seviezhou/faust/compiler/libcode.cpp:197
    #19 0xba40ae in __asan::AsanThread::ThreadStart(unsigned long, __sanitizer::atomic_uintptr_t*) (/home/seviezhou/faust/build/bin/faust+0xba40ae)
    #20 0x7f2f749d96da in start_thread (/lib/x86_64-linux-gnu/libpthread.so.0+0x76da)
    #21 0x7f2f73d4088e in clone /build/glibc-OTsEL5/glibc-2.27/misc/../sysdeps/unix/sysv/linux/x86_64/clone.S:95

AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/seviezhou/faust/compiler/signals/signals.hh:266:18 in isNum(CTree* const&, num&)
Thread T1 created by T0 here:
    #0 0xaef650 in pthread_create (/home/seviezhou/faust/build/bin/faust+0xaef650)
    #1 0xcbb55e in callFun(void* (*)(void*)) /home/seviezhou/faust/compiler/libcode.cpp:186:5
    #2 0xc8d86d in compileFaustFactoryAux(int, char const**, char const*, char const*, bool) /home/seviezhou/faust/compiler/libcode.cpp:1909:5
    #3 0xc8c854 in compileFaustFactory(int, char const**, char const*, char const*, std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >&, bool)
/home/seviezhou/faust/compiler/libcode.cpp:1982:9
    #4 0xccc68d in main /home/seviezhou/faust/compiler/main.cpp:45:33
    #5 0x7f2f73c40b96 in __libc_start_main /build/glibc-OTsEL5/glibc-2.27/csu/../csu/libc-start.c:310

==14194==ABORTING
```

POC

SEGV-computeSigOutput-cosprim-49.zip

sletz commented on Aug 26, 2020 Contributor

Thanks. What is the actual compiled DSP source code?

seviezhou commented on Aug 26, 2020 Author

I put the POC in the attached file, it is something like:

```
// check removed from the code
process =!(int :>int), float, float(hslider("cos", 0, 0, 10, 1));
```

It might not be valid, and is produced by random mutation.

sletz commented on Aug 26, 2020 Contributor

Thanks, this is indeed a known problem when language keywords are use in labels.

Out of curiosity: what is this "produced by random mutation" idea or project? Thanks.

seviezhou commented on Aug 26, 2020 Author

It is produced by the fuzzing technique, the most popular tool implementing such technique is AFL.

sletz commented on Aug 26, 2020 Contributor

Interesting. Do you actually use APL to test Faust? Or any other fuzzing tool? In any case assuming this is public, I would be interested to see the code.

seviezhou commented on Aug 26, 2020 Author

I use my own tool, and it is not currently public available. But my tool shares similar mutation operations with AFL, you can just read the code of AFL if you are interested in it.

sletz mentioned this issue on Dec 3, 2020

process = hslider("min", 0, 0, 1, 0.1); crashes compiler #527

Closed

sletz mentioned this issue on Jun 24, 2021

AddressSanitizer: SEGV on unknown address 0x000000000000 #604

Open

kirotawa commented on Oct 18, 2021

CVE-2021-32275 was assigned to this issue.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

3 participants