heap-buffer-overflow in SEIUnit::mvc_scalable_nesting #427

Closed **cemonatk** opened this issue on May 24, 2021 · 0 comments

Labels bug

cemonatk commented on May 24, 2021

Hi, please see asan output and poc file below.

Found by **Cem Onat Karagun of Diesec**

System info:

```
Ubuntu 21.04
tsMuxeR version git-f6ab2a2
```

To run PoC after unzip:

```
$ ./tsmuxer scalable_nesting_poc
```

scalable_nesting_poc.zip

Asan output:

```
tsMuxeR version git-f6ab2a2. github.com/justdan96/tsMuxer
=================================================================
==3847316==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x60200000d517 at pc 0x000000302336 bp 0x7ffcadc46bc0 sp 0x7ffcadc46368
READ of size 16 at 0x60200000d517 thread T0
    #0 0x302335 in MemcmpInterceptorCommon(void*, int (*)(void const*, void const*, unsigned long), void const*, void const*, unsigned long)
(/home/Fuzzer_Instance_11/txmux/tsMuxer/bin/tsMuxeR+0x302335)
    #1 0x30282a in memcmp (/home/Fuzzer_Instance_11/txmux/tsMuxer/bin/tsMuxeR+0x30282a)
    #2 0x78171c in SEIUnit::mvc_scalable_nesting(SPSUnit&, unsigned char*, int, int) /src/build/../tsMuxer/nalUnits.cpp:2442:21
    #3 0x77edc9 in SEIUnit::deserialize(SPSUnit&, int) /src/build/../tsMuxer/nalUnits.cpp:2015:13
    #4 0x4e3408 in H264StreamReader::checkStream(unsigned char*, int) /src/build/../tsMuxer/h264StreamReader.cpp:142:25
    #5 0x6ceacc in METADemuxer::detectTrackReader(unsigned char*, int, AbstractStreamReader::ContainerType, int, int) /src/build/../tsMuxer/metaDemuxer.cpp:745:21
    #6 0x6c7255 in METADemuxer::DetectStreamReader(BufferedReaderManager&, std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> > const&, bool)
/src/build/../tsMuxer/metaDemuxer.cpp:684:35
    #7 0x5df87e in detectStreamReader(char const*, MPLSParser*, bool) /src/build/../tsMuxer/main.cpp:120:34
    #8 0x5efd05 in main /src/build/../tsMuxer/main.cpp:698:17
    #9 0x7f3c2da0d564 in __libc_start_main csu/../csu/libc-start.c:332:16
    #10 0x2ebded in _start (/home/Fuzzer_Instance_11/txmux/tsMuxer/bin/tsMuxeR+0x2ebded)

0x60200000d517 is located 0 bytes to the right of 7-byte region [0x60200000d510,0x60200000d517)
allocated by thread T0 here:
    #0 0x39823d in operator new[](unsigned long) (/home/Fuzzer_Instance_11/txmux/tsMuxer/bin/tsMuxeR+0x39823d)
    #1 0x74f45d in NALUnit::decodeBuffer(unsigned char const*, unsigned char const*) /src/build/../tsMuxer/nalUnits.cpp:282:19

SUMMARY: AddressSanitizer: heap-buffer-overflow (/home/Fuzzer_Instance_11/txmux/tsMuxer/bin/tsMuxeR+0x302335) in MemcmpInterceptorCommon(void*, int (*)(void const*, void const*,
unsigned long), void const*, void const*, unsigned long)
Shadow bytes around the buggy address:
  0x0c047fff9a50: fa fa 00 00 fa fa 00 00 fa fa 00 00 fa fa 00 00
  0x0c047fff9a60: fa fa 00 00 fa fa 00 00 fa fa 00 00 fa fa 00 00
  0x0c047fff9a70: fa fa 00 00 fa fa 00 00 fa fa 00 00 fa fa 00 00
  0x0c047fff9a80: fa fa 00 00 fa fa 01 fa fa fa 04 fa fa fa 04 fa
  0x0c047fff9a90: fa fa 01 fa fa fa 04 fa fa fa 04 fa fa fa 00 00
=>0x0c047fff9aa0: fa fa[07]fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff9ab0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff9ac0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff9ad0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff9ae0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff9af0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
  Left alloca redzone:     ca
  Right alloca redzone:    cb
  Shadow gap:              cc
==3847316==ABORTING
```

jcdr428 mentioned this issue on May 24, 2021

[bug] heap buffer overflow when last byte of SEI = 0xFF #425

Merged

xavery closed this as completed in `ea879f3` on Jun 9, 2021

jcdr428 added the bug label on Jun 23

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants