New issue

Jump to bottom

heap-buffer-overflow in elf_reader #763

Closed CCWANG19 opened this issue on Aug 9 · 1 comment

Assignees 

CCWANG19 commented on Aug 9

version

```
latest master [5d1d643](https://github.com/lief-project/LIEF/commit/5d1d643a0c46437b57d35654690e03d26d189070)
```

Build platform

Ubuntu 20.04.3 LTS (Linux 5.13.0-52-generic x86_64)

Build step

```
cmake .. -DCMAKE_CXX_FLAGS="-fsanitize=address -g" -DCMAKE_C_FLAGS="-fsanitize=address -g" -DCMAKE_EXE_LINKER_FLAGS="-fsanitize=address"
```

Run

```
./build/examples/c/elf_reader poc
```

poc.zip

AddressSanitizer output

```
Can't access the content of section #0
Can't access the content of section #1
Can't access the content of section #2
```

```
=================================================================
==2292604==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x60d000000268 at pc
0x7f5b35470a7d bp 0x7ffe10d8f4c0 sp 0x7ffe10d8ec68
READ of size 109 at 0x60d000000268 thread T0
    #0 0x7f5b35470a7c in __interceptor_strlen
../../../../src/libsanitizer/sanitizer_common/sanitizer_common_interceptors.inc:354
    #1 0x55a087822aa7 in std::char_traits<char>::length(char const*)
/usr/include/c++/9/bits/char_traits.h:342
    #2 0x55a087822aa7 in std::__cxx11::basic_string<char, std::char_traits<char>,
std::allocator<char> >::assign(char const*) /usr/include/c++/9/bits/basic_string.h:1443
    #3 0x55a087822aa7 in std::__cxx11::basic_string<char, std::char_traits<char>,
std::allocator<char> >::operator=(char const*) /usr/include/c++/9/bits/basic_string.h:709
    #4 0x55a087822aa7 in void LIEF::ELF::CorePrPsInfo::parse_<LIEF::ELF::details::ELF32>()
/home/wcc/LIEF/src/ELF/NoteDetails/core/CorePrPsInfo.tcc:41
    #5 0x55a087822aa7 in LIEF::ELF::CorePrPsInfo::parse()
/home/wcc/LIEF/src/ELF/NoteDetails/core/CorePrPsInfo.cpp:156
    #6 0x55a0878260de in LIEF::ELF::CorePrPsInfo::make(LIEF::ELF::Note&)
/home/wcc/LIEF/src/ELF/NoteDetails/core/CorePrPsInfo.cpp:44
    #7 0x55a0877383f3 in LIEF::ELF::Note::details() /home/wcc/LIEF/src/ELF/Note.cpp:150
    #8 0x55a08773bbe0 in LIEF::ELF::Note::Note(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&, LIEF::ELF::NOTE_TYPES_CORE,
std::vector<unsigned char, std::allocator<unsigned char> > const&, LIEF::ELF::Binary*)
/home/wcc/LIEF/src/ELF/Note.cpp:92
    #9 0x55a08767077d in std::_MakeUniq<LIEF::ELF::Note>::__single_object
std::make_unique<LIEF::ELF::Note, std::__cxx11::basic_string<char, std::char_traits<char>,
std::allocator<char> >&, LIEF::ELF::NOTE_TYPES_CORE, std::vector<unsigned char,
std::allocator<unsigned char> >, LIEF::ELF::Binary*>(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> >&, LIEF::ELF::NOTE_TYPES_CORE&&,
std::vector<unsigned char, std::allocator<unsigned char> >&&, LIEF::ELF::Binary*&&)
/usr/include/c++/9/bits/unique_ptr.h:857
    #10 0x55a08767077d in LIEF::ELF::Parser::parse_notes(unsigned long, unsigned long)
/home/wcc/LIEF/src/ELF/Parser.cpp:568
    #11 0x55a0876ec34e in boost::leaf::result<LIEF::ok_t>
LIEF::ELF::Parser::parse_binary<LIEF::ELF::details::ELF32>() /home/wcc/LIEF/src/ELF/Parser.tcc:296
    #12 0x55a08767371e in LIEF::ELF::Parser::init(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&) /home/wcc/LIEF/src/ELF/Parser.cpp:323
    #13 0x55a087674f03 in LIEF::ELF::Parser::parse(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&, LIEF::ELF::DYNSYM_COUNT_METHODS)
/home/wcc/LIEF/src/ELF/Parser.cpp:342
    #14 0x55a087383c06 in elf_parse /home/wcc/LIEF/api/c/ELF/Binary.cpp:67
    #15 0x55a087342c96 in main /home/wcc/LIEF/examples/c/elf_reader.c:16
    #16 0x7f5b34eef0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x240b2)
    #17 0x55a08738373d in _start (/home/wcc/LIEF/build/examples/c/elf_reader+0x31c73d)

0x60d000000268 is located 0 bytes to the right of 136-byte region [0x60d0000001e0,0x60d000000268)
allocated by thread T0 here:
    #0 0x7f5b35518587 in operator new(unsigned long)
../../../../src/libsanitizer/asan/asan_new_delete.cc:104
    #1 0x55a08773b79b in __gnu_cxx::new_allocator<unsigned char>::allocate(unsigned long, void
const*) /usr/include/c++/9/ext/new_allocator.h:114
    #2 0x55a08773b79b in std::allocator_traits<std::allocator<unsigned char>
>::allocate(std::allocator<unsigned char>&, unsigned long)
/usr/include/c++/9/bits/alloc_traits.h:443
    #3 0x55a08773b79b in std::_Vector_base<unsigned char, std::allocator<unsigned char>
>::_M_allocate(unsigned long) /usr/include/c++/9/bits/stl_vector.h:343
    #4 0x55a08773b79b in std::_Vector_base<unsigned char, std::allocator<unsigned char>
```

```
>::_M_create_storage(unsigned long) /usr/include/c++/9/bits/stl_vector.h:358
    #5 0x55a08773b79b in std::_Vector_base<unsigned char, std::allocator<unsigned char>
>::_Vector_base(unsigned long, std::allocator<unsigned char> const&)
/usr/include/c++/9/bits/stl_vector.h:302
    #6 0x55a08773b79b in std::vector<unsigned char, std::allocator<unsigned char>
>::vector(std::vector<unsigned char, std::allocator<unsigned char> > const&)
/usr/include/c++/9/bits/stl_vector.h:552
    #7 0x55a08773b79b in LIEF::ELF::Note::Note(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&, LIEF::ELF::NOTE_TYPES_CORE,
std::vector<unsigned char, std::allocator<unsigned char> > const&, LIEF::ELF::Binary*)
/home/wcc/LIEF/src/ELF/Note.cpp:89

SUMMARY: AddressSanitizer: heap-buffer-overflow
../../../../src/libsanitizer/sanitizer_common/sanitizer_common_interceptors.inc:354 in
__interceptor_strlen
Shadow bytes around the buggy address:
  0x0c1a7fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c1a7fff8000: fa fa fa fa fa fa fa fa 00 00 00 00 00 00 00 00
  0x0c1a7fff8010: 00 00 00 00 00 00 00 00 00 fa fa fa fa fa fa fa
  0x0c1a7fff8020: fa fa 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c1a7fff8030: 00 00 00 fa fa fa fa fa fa fa fa fa 00 00 00 00
=>0x0c1a7fff8040: 00 00 00 00 00 00 00 00 00 00 00 00 00[fa]fa fa
  0x0c1a7fff8050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c1a7fff8060: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c1a7fff8070: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c1a7fff8080: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c1a7fff8090: 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
==2292604==ABORTING
```

CCWANG19 assigned romainthomas on Aug 9

romainthomas closed this as completed in `53bf680` on Aug 10

---

romainthomas commented on Aug 10 — Member

Thank you **@CCWANG19** for reporting these issues!

whyitfor mentioned this issue on Oct 3

Bump lief to 0.12.2 redballoonsecurity/ofrak#47

Merged

romainthomas added a commit that referenced this issue 25 days ago

`Fix #763` `45f3c77`

Assignees

romainthomas

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants