New issue

Jump to bottom

SEGV in LIEF::MachO::SegmentCommand::virtual_address at MachO/SegmentCommand.cpp:137 #784

Closed **bladchan** opened this issue on Sep 11 · 4 comments

Assignees

Labels bug MachO Parser

bladchan commented on Sep 11

Describe the bug
A bad macho file which can lead LIEF::MachO::Parser::parse() to segmentation fault.
Poc is here : poc2.zip

To Reproduce

1. Build the whole project with **ASAN**
2. Drive program (compile it with **ASAN** too):

```
// read_mecho.c
#include <LIEF/LIEF.hpp>

int main(int argc, char** argv){

        if(argc != 2) return 0;

        try {
            std::unique_ptr<LIEF::MachO::FatBinary> macho = LIEF::MachO::Parser::parse(argv[1]);
        } catch (const LIEF::exception& err) {
            std::cerr << err.what() << std::endl;
        }

        return 0;
}
```

3. Run Poc:

```
$ ./read_macho ./poc2.bin
```

Expected behavior
Parse the Mach-O file without segmentation fault because segmentation fault can cause a Denial of Service (Dos).

Environment (please complete the following information):

- System and Version : Ubuntu 20.04 + gcc 9.4.0
- Target format : **Mach-O**
- LIEF commit version: `ad81191`

Additional context
ASAN says:

```
ubuntu@ubuntu:~/test/LIEF/fuzz$ ./read_macho ./poc2.bin
nlist[0].str_idx seems corrupted (0x00700000)
nlist[1].str_idx seems corrupted (0x00015381)
.......
Indirect symbol index is out of range (1392508928 vs max sym: 356)
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
Wrong index: 7
AddressSanitizer:DEADLYSIGNAL
=================================================================
==837035==ERROR: AddressSanitizer: SEGV on unknown address 0x000000000058 (pc 0x5653a24eacfd bp
0x7ffdbf694b60 sp 0x7ffdbf6943d0 T0)
==837035==The signal is caused by a READ memory access.
==837035==Hint: address points to the zero page.
    #0 0x5653a24eacfc in LIEF::MachO::SegmentCommand::virtual_address() const
/home/ubuntu/test/LIEF/src/MachO/SegmentCommand.cpp:137
    #1 0x5653a22244b8 in boost::leaf::result<LIEF::ok_t>
LIEF::MachO::BinaryParser::parse_dyldinfo_generic_bind<LIEF::MachO::details::MachO32>()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.tcc:1631
    #2 0x5653a21f1a79 in boost::leaf::result<LIEF::ok_t>
LIEF::MachO::BinaryParser::parse_dyldinfo_binds<LIEF::MachO::details::MachO32>()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.tcc:1357
    #3 0x5653a21c1735 in boost::leaf::result<LIEF::ok_t>
LIEF::MachO::BinaryParser::parse<LIEF::MachO::details::MachO32>()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.tcc:113
    #4 0x5653a21b2348 in LIEF::MachO::BinaryParser::init_and_parse()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.cpp:145
    #5 0x5653a21b1ab0 in LIEF::MachO::BinaryParser::parse(std::unique_ptr<LIEF::BinaryStream,
std::default_delete<LIEF::BinaryStream> >, unsigned long, LIEF::MachO::ParserConfig const&)
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.cpp:125
    #6 0x5653a1a3bc01 in LIEF::MachO::Parser::build()
```

```
/home/ubuntu/test/LIEF/src/MachO/Parser.cpp:174
    #7 0x5653a1a38995 in LIEF::MachO::Parser::parse(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&, LIEF::MachO::ParserConfig const&)
/home/ubuntu/test/LIEF/src/MachO/Parser.cpp:64
    #8 0x5653a18a3923 in main /home/ubuntu/test/LIEF/fuzz/read_macho.c:8
    #9 0x7f5206270082 in __libc_start_main ../csu/libc-start.c:308
    #10 0x5653a18a355d in _start (/home/ubuntu/test/LIEF/fuzz/read_macho+0x33055d)

AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/ubuntu/test/LIEF/src/MachO/SegmentCommand.cpp:137 in
LIEF::MachO::SegmentCommand::virtual_address() const
==837035==ABORTING
```

Hope that helps!

bladchan assigned **romainthomas** on Sep 11

romainthomas added **MachO** bug Parser labels on Sep 11

bladchan commented on Sep 11 — Author

There is another bad macho file which can lead LIEF::MachO::Parser::parse() to segmentation fault. Maybe it is the same reason which caused segmentation fault in MachO/SegmentCommand.cpp, so I report it under this issue.

Poc here : poc3.zip

ASAN says:

```
ubuntu@ubuntu:~/test/LIEF/fuzz$ ./read_macho poc3.bin
Segment __TEXT: content corrupted!
Unknown architecture
Command 'Out of range' not parsed!
Can't read the raw data of the load command
AddressSanitizer:DEADLYSIGNAL
=================================================================
==428943==ERROR: AddressSanitizer: SEGV on unknown address 0x000000000068 (pc 0x557a1c3e0e70 bp
0x0fffa48dad9a sp 0x7ffd246d6c80 T0)
==428943==The signal is caused by a READ memory access.
==428943==Hint: address points to the zero page.
    #0 0x557a1c3e0e6f in LIEF::MachO::SegmentCommand::file_offset() const
/home/ubuntu/test/LIEF/src/MachO/SegmentCommand.cpp:149
    #1 0x557a1bfa05d9 in LIEF::MachO::Binary::segment_from_offset(unsigned long) const
/home/ubuntu/test/LIEF/src/MachO/Binary.cpp:545
    #2 0x557a1bfa0c28 in LIEF::MachO::Binary::segment_from_offset(unsigned long)
/home/ubuntu/test/LIEF/src/MachO/Binary.cpp:573
    #3 0x557a1c0ee530 in boost::leaf::result<LIEF::ok_t>
LIEF::MachO::BinaryParser::post_process<LIEF::MachO::details::MachO32>
```

```
(LIEF::MachO::CodeSignature&) /home/ubuntu/test/LIEF/src/MachO/BinaryParser.tcc:3397
    #4 0x557a1c0b8646 in boost::leaf::result<LIEF::ok_t>
LIEF::MachO::BinaryParser::parse<LIEF::MachO::details::MachO32>()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.tcc:168
    #5 0x557a1c0a8348 in LIEF::MachO::BinaryParser::init_and_parse()
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.cpp:145
    #6 0x557a1c0a7ab0 in LIEF::MachO::BinaryParser::parse(std::unique_ptr<LIEF::BinaryStream,
std::default_delete<LIEF::BinaryStream> >, unsigned long, LIEF::MachO::ParserConfig const&)
/home/ubuntu/test/LIEF/src/MachO/BinaryParser.cpp:125
    #7 0x557a1b931c01 in LIEF::MachO::Parser::build()
/home/ubuntu/test/LIEF/src/MachO/Parser.cpp:174
    #8 0x557a1b92e995 in LIEF::MachO::Parser::parse(std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> > const&, LIEF::MachO::ParserConfig const&)
/home/ubuntu/test/LIEF/src/MachO/Parser.cpp:64
    #9 0x557a1b799923 in main /home/ubuntu/test/LIEF/fuzz/read_macho.c:8
    #10 0x7fac58975082 in __libc_start_main ../csu/libc-start.c:308
    #11 0x557a1b79955d in _start (/home/ubuntu/test/LIEF/fuzz/read_macho+0x33055d)

AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/ubuntu/test/LIEF/src/MachO/SegmentCommand.cpp:149 in
LIEF::MachO::SegmentCommand::file_offset() const
==428943==ABORTING
```

romainthomas commented on Sep 11 — Member

Thank you very much **@bladchan** for these issues. I'm looking at them

romainthomas closed this as completed in `24935f6` on Sep 12

---

rusNaN12 commented on Oct 20

Do we have a release version for this fix?

romainthomas commented on Oct 21 — Member

I plan to release a patch version for LIEF (`0.12.3`) along with the release of Python 3.11

romainthomas added a commit that referenced this issue 25 days ago

`Fix #784` `ddc77e2`

Assignees

romainthomas

Labels

bug MachO Parser

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

3 participants