New issue

Jump to bottom

A heap-buffer-overflow in maxminddb.c:2019:13 #236

Closed **seviezhou** opened this issue on Aug 4, 2020 · 4 comments · Fixed by #237

seviezhou commented on Aug 4, 2020

System info

Ubuntu X64, gcc (Ubuntu 5.5.0-12ubuntu1), mmdblookup (latest master e6e63a)

Configure

CFLAGS="-g -fsanitize=address" LDFLAGS="-fsanitize=address" ./configure --enable-static

Command line

./bin/.libs/lt-mmdblookup --ip 127.0.0.1 --file @@

AddressSanitizer output

```
=================================================================
==4648==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6020000000d1 at pc 0x0000004480b1 bp 0x7ffd0e2ccb00 sp 0x7ffd0e2cc2b0
READ of size 2 at 0x6020000000d1 thread T0
    #0 0x4480b0 in printf_common(void*, char const*, __va_list_tag*) /home/seviezhou/llvm-6.0.0/projects/compiler-rt/lib/asan/../sanitizer_common/sanitizer_common_interceptors_format.inc:548
    #1 0x448b2a in __interceptor_vfprintf /home/seviezhou/llvm-6.0.0/projects/compiler-rt/lib/asan/../sanitizer_common/sanitizer_common_interceptors.inc:1549
    #2 0x448be2 in __interceptor_fprintf /home/seviezhou/llvm-6.0.0/projects/compiler-rt/lib/asan/../sanitizer_common/sanitizer_common_interceptors.inc:1606
    #3 0x7fe1bea26a5f in dump_entry_data_list /home/seviezhou/libmaxminddb/src/maxminddb.c:2019:13
    #4 0x7fe1bea24c8f in MMDB_dump_entry_data_list /home/seviezhou/libmaxminddb/src/maxminddb.c:1917:5
    #5 0x519c9f in lookup_and_print /home/seviezhou/libmaxminddb/bin/mmdblookup.c:526:13
    #6 0x519498 in main /home/seviezhou/libmaxminddb/bin/mmdblookup.c:134:14
    #7 0x7fe1bdb2183f in __libc_start_main /build/glibc-e6zv40/glibc-2.23/csu/../csu/libc-start.c:291
    #8 0x41a808 in _start (/home/seviezhou/libmaxminddb/bin/.libs/lt-mmdblookup+0x41a808)

0x6020000000d1 is located 0 bytes to the right of 1-byte region [0x6020000000d0,0x6020000000d1)
allocated by thread T0 here:
    #0 0x4dea18 in __interceptor_malloc /home/seviezhou/llvm-6.0.0/projects/compiler-rt/lib/asan/asan_malloc_linux.cc:88
    #1 0x7fe1bea2508f in bytes_to_hex /home/seviezhou/libmaxminddb/src/maxminddb.c:2106:18
    #2 0x7fe1bea2508f in dump_entry_data_list /home/seviezhou/libmaxminddb/src/maxminddb.c:2011
    #3 0x7fe1bea24c8f in MMDB_dump_entry_data_list /home/seviezhou/libmaxminddb/src/maxminddb.c:1917:5
    #4 0x519c9f in lookup_and_print /home/seviezhou/libmaxminddb/bin/mmdblookup.c:526:13
    #5 0x519498 in main /home/seviezhou/libmaxminddb/bin/mmdblookup.c:134:14
    #6 0x7fe1bdb2183f in __libc_start_main /build/glibc-e6zv40/glibc-2.23/csu/../csu/libc-start.c:291

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/seviezhou/llvm-6.0.0/projects/compiler-rt/lib/asan/../sanitizer_common/sanitizer_common_interceptors_format.inc:548 in printf_common(void*, char const*, __va_list_tag*)
Shadow bytes around the buggy address:
  0x0c047fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff8000: fa fa 00 fa fa fa 00 fa fa fa 03 fa fa fa 00 fa
=>0x0c047fff8010: fa fa 00 00 fa fa 03 fa fa fa[01]fa fa fa fa fa
  0x0c047fff8020: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8030: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8060: 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
==4648==ABORTING
```

POC

heap-overflow-dump_entry_data_list-maxminddb-2019.zip

oschwald commented on Aug 4, 2020 Member

Thanks! Nice find. Which fuzzer did you use to find this?

seviezhou commented on Aug 4, 2020 Author

It is a modified version of AFL. I think you can use AFL to test your code, because AFL can also find such bug.

👍 1

oschwald added a commit that referenced this issue on Aug 5, 2020

```
Replace most malloc uses with calloc   …                                  2b752c3
```

oschwald mentioned this issue on Aug 5, 2020

Fix heap buffer overflow #237

Merged

oschwald added a commit that referenced this issue on Aug 5, 2020

```
Replace most malloc uses with calloc   …                                  eac45e2
```

horgh closed this as completed in #237 on Aug 6, 2020

oschwald commented on Aug 6, 2020 Member

1.4.3 has been released with a fix for this.

rfrohl commented on Nov 6, 2020

the issue got CVE-2020-28241 assigned

👍 2

Assignees

No one assigned

Labels

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

Fix heap buffer overflow
maxmind/libmaxminddb

3 participants