New issue

Jump to bottom

Heap-buffer-overflow found at mms_client_example1.c different from issues 6 #7

Open **Rrooach** opened this issue on Oct 10, 2019 · 0 comments

Rrooach commented on Oct 10, 2019

Hello, I found a potential heap-buffer-overflow in /libiec_iccp_mod/examples/mms_client_example1/mms_client_example1.c, but unable to locate the trace code, however the deeper cause is found in: /root/libiec_iccp_mod/src/mms/iso_mms/client/mms_client_connection.c:790
when the program try to calloc a large spcae, caused the heap-buffer-overflow.

Below are steps followed to reproduce crash
Download latest source code from: /fcovatti/libiec_iccp_mod/, compiled with clang and ASAN `export CFLAGS="-g -fsanitize=address" LDFLAGS="-fsanitize=address"` before make

Row data
crash.zip

ASAN Output:

```
=================================================================
==23235==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6280000040ff at pc 0x00000048e36f bp 0x7ffef31f51d0 sp 0x7ffef31f4980   WRITE of size 127 at 0x6280000040ff
thread T0                                                                                                                                  #12 0x7fecd633e82f in __libc_start_main /build/glibc-LK5gWL/glibc-
2.23/csu/../csu/libc-start.c:291
    #13 0x41a1c8 in _start (/root/temp/iec/libiec_iccp_mod/examples/mms_client_example1/mms_client_example1+0x41a1c8)

0x6280000040ff is located 1 bytes to the left of 16100-byte region [0x628000004100,0x628000007fe4)                                  allocated by thread T0 here:
    #0 0x4daee0 in calloc (/root/temp/iec/libiec_iccp_mod/examples/mms_client_example1/mms_client_example1+0x4daee0)
    #1 0x51443c in MmsConnection_create /root/libiec_iccp_mod/src/mms/iso_mms/client/mms_client_connection.c:790
    #2 0x7fecd633e82f in __libc_start_main /build/glibc-LK5gWL/glibc-2.23/csu/../csu/libc-start.c:291

SUMMARY: AddressSanitizer: heap-buffer-overflow (/root/temp/iec/libiec_iccp_mod/examples/mms_client_example1/mms_client_example1+0x48e36e) in read
Shadow bytes around the buggy address:
0x0c507fff87c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff87d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff87e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff87f0: 00 00 00 00 00 00 00 00 00 00 00 00 04 fa fa fa
0x0c507fff8800: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c507fff8810: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa[fa]                                                                    0
x0c507fff8820: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff8830: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
840: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff8850: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c507fff8860: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
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
==23235==ABORTING
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant