New issue

Jump to bottom

A stackoverflow bug #374

Closed **WaterDemo** opened this issue on Jun 1 · 3 comments

Assignees

WaterDemo commented on Jun 1 • edited

Hi, there is a stack overflow bug in OpENer, which is found by fuzzing.

It can be reproduced by the poc.zip attached (by using `send_testcase.py` script and poc.zip is assumed to be unziped).

poc.zip

Here is the message output by AddressSanitizer:

==210407==ERROR: AddressSanitizer: stack-buffer-overflow on address 0x7fffa249f590 at pc 0x00000056073e bp 0x7fffa249e890 sp 0x7fffa249e888
READ of size 1 at 0x7fffa249f590 thread T0
#0 0x56073d (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x56073d)
#1 0x536d27 (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x536d27)
#2 0x52e8ab (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x52e8ab)
#3 0x54da0c (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x54da0c)
#4 0x558400 (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x558400)
#5 0x55d6b0 (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x55d6b0)
#6 0x55c5d7 (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x55c5d7)
#7 0x5688eb (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x5688eb)
#8 0x56638b (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x56638b)
#9 0x52e24e (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x52e24e)
#10 0x52e0ed (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x52e0ed)
#11 0x7f904081e0b2 (/lib/x86_64-linux-gnu/libc.so.6+0x240b2)
#12 0x42750d (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x42750d)

Address 0x7fffa249f590 is located in stack of thread T0 at offset 560 in frame
#0 0x567fcf (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x567fcf)

This frame has 7 object(s):
[32, 36) 'remaining_bytes'
[48, 560) 'incoming_message' <== Memory access at offset 560 overflows this variable
[624, 632) 'read_buffer'
[656, 672) 'sender_address'
[688, 692) 'fromlen'
[704, 712) 'agg.tmp'
[736, 1264) 'outgoing_message'
HINT: this may be a false positive if your program uses some custom stack unwind mechanism or swapcontext
(longjmp and C++ exceptions *are* supported)
SUMMARY: AddressSanitizer: stack-buffer-overflow (/home/OpENer/bin/posix/src/ports/POSIX/OpENer+0x56073d)
Shadow bytes around the buggy address:
0x10007448be60: 00 00 00 00 00 00 00 00 00 00 00 00 f1 f1 f1 f1
0x10007448be70: 04 f2 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448be80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448be90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448bea0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x10007448beb0: 00 00[f2]f2 f2 f2 f2 f2 f2 f2 00 f2 f2 f2 00 00
0x10007448bec0: f2 f2 04 f2 00 f2 f2 f2 00 00 00 00 00 00 00 00
0x10007448bed0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448bee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448bef0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10007448bf00: 00 00 00 00 00 00 00 00 00 00 f3 f3 f3 f3 f3 f3
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
==210407==ABORTING

WaterDemo changed the title ~~a stackoverflow bug found by fuzzing~~ **A stackoverflow bug** on Jun 1

CapXilinx commented on Jun 14 — Member

Thanks, will look into this asap

CapXilinx self-assigned this on Jun 14

WaterDemo closed this as completed on Jul 15

CapXilinx commented on Jul 16 — Member

Hi **@WaterDemo**
May I ask if this has resolved in the meantime?
Didnt hat the time to look into this up to now unfortunately

WaterDemo commented on Jul 16 — Author

Hi, not yet, should I Reopen it?

Assignees

CapXilinx

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants