huntr

Use After Free in vim/vim

✓ Valid Reported on Oct 24th 2021

0

Chat with us

Description

Greetings,

A Use After Free issue was discovered in Vim.

The POC file is reduced to the absolute minimum to reproduce the problem. Please see sanitizer output and the "trimmed" POC file link below.

System info OS version : Ubuntu 20.04.2 LTS + Clang 12 with ASan Vim Version : master(3c5904d) - Sun Oct 24 14:50:07 2021 +0100

Steps to reproduce:

```
git clone https://github.com/vim/vim
```

```
LD=lld-12 AS=llvm-as-12 AR=llvm-ar-12 RANLIB=llvm-ranlib-12 CC=clang-12 CX
```

Download POC from This URL

```
./vim -u NONE -X -Z -e -s -S POC -c :qa!
```

Sanitizer output:

```
=================================================================
==126137==ERROR: AddressSanitizer: heap-use-after-free on address 0x602000
READ of size 1 at 0x602000006dd1 thread T0
    #0 0x121b06a in nfa_regmatch /src/fuzzer11/triage_yeni/vim/src/./regexp
    #1 0x11f4062 in nfa_regtry /src/fuzzer11/triage_yeni/vim/src/./regexp_r
    #2 0x11f4062 in nfa_regexec_both /src/fuzzer11/triage_yeni/vim/src/./re
    #3 0x113f156 in vim_regexec_multi /src/fuzzer11/triage_yeni/vim/src/reg
    #4 0x9163de in ex_substitute /src/fuzzer11/triage_yeni/vim/src/ex_cmds.
    #5 0x94ff7b in do_one_cmd /src/fuzzer11/triage_yeni/vim/src/ex_docmd.c:
    #6 0x94ff7b in do_cmdline /src/fuzzer11/triage_yeni/vim/src/ex_docmd.c:
    #7 0x136cde4 in do_source /src/fuzzer11/triage_yeni/vim/src/scriptfile.
    #8 0x13699e1 in cmd_source /src/fuzzer11/triage_yeni/vim/src/scriptfile
    #9 0x13699e1 in ex_source /src/fuzzer11/triage_yeni/vim/src/scriptfile.
    #10 0x94ff7b in do_one_cmd /src/fuzzer11/triage_yeni/vim/src/ex_docmd.c
    #11 0x94ff7b in do_cmdline /src/fuzzer11/triage_yeni/vim/src/ex_docmd.c
    #12 0x1bcecfc in exe_commands /src/fuzzer11/triage_yeni/vim/src/main.c:
    #13 0x1bcecfc in vim_main2 /src/fuzzer11/triage_yeni/vim/src/main.c:77
    #14 0x1bc5a8f in main /src/fuzzer11/triage_yeni/vim/src/main.c:425:12
    #15 0x7fc9c677c0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.
    #16 0x41f64d in _start (/src/fuzzer11/triage_yeni/vim/src/vim+0x41f64d)

0x602000006dd1 is located 1 bytes inside of 10-byte region [0x602000006dd0,
freed by thread T0 here:
    #0 0x49a642 in free (/src/fuzzer11/triage_yeni/vim/src/vim+0x49a642)
    #1 0xd5ee9f in ml_flush_line /src/fuzzer11/triage_yeni/vim/src/memline.

previously allocated by thread T0 here:
    #0 0x49a8ad in malloc (/src/fuzzer11/triage_yeni/vim/src/vim+0x49a8ad)
    #1 0x4cc2cb in lalloc /src/fuzzer11/triage_yeni/vim/src/alloc.c:244:11

SUMMARY: AddressSanitizer: heap-use-after-free /src/fuzzer11/triage_yeni/vi
Shadow bytes around the buggy address:
  0x0c047fff8d60: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8d70: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8d80: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8d90: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8da0: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
=>0x0c047fff8db0: fa fa fd fd fa fa fd fd fa fa[fd]fd fa fa 00 05
  0x0c047fff8dc0: fa fa 02 fa fa fa 00 03 fa fa fd fa fa fa fd fa
  0x0c047fff8dd0: fa fa 00 03 fa fa 00 03 fa fa 01 fa fa fa 05 fa
  0x0c047fff8de0: fa fa 05 fa fa fa 05 fa fa fa 01 fa fa fa fa fa
  0x0c047fff8df0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8e00: 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
==126137==ABORTING
```

References:

CWE-416: Use After Free - https://cwe.mitre.org/data/definitions/416.html

This vulnerability is capable of crashing software, bypass protection mechanism, modify of
nd successful exploitation may lead to code execution

nces

- Cem Onat Karagun

(Published)

Vulnerability Type
CWE-416: Use After Free

Severity
High (7.3)

Affected Version
*

Visibility
Public

Status
Fixed

Found by

cem
@cemonatk
unranked

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 888 times.

We have contacted a member of the **vim** team and are waiting to hear back a year ago

We have sent a follow up to the **vim** team. We will try again in 7 days. a year ago

We have sent a second follow up to the **vim** team. We will try again in 10 days. a year ago

We have sent a third and final follow up to the **vim** team. This report is now considered stale. a year ago

Bram Moolenaar a year ago

Sorry for the delay, I was busy with other things.
I can reproduce the use-after-free

Bram Moolenaar validated this vulnerability a year ago

cem has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

Bram Moolenaar a year ago

Fix is in patch 8.2.3612, please verify

cem a year ago Researcher

Hi Bram, thanks for the fixes. I'll check them all asap.

cem a year ago Researcher

Looks good - tested with patch 8.2.3616.

Bram Moolenaar marked this as fixed with commit **64066b** a year ago

Bram Moolenaar has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

Jamie Slome a year ago Admin

CVE published! 🎊

Jamie Slome a year ago Admin

@cemonatk 👋 - same situation here! A bug caused the reward to erroneously be set to $355. I have reset the reward to the one shown at the point of disclosure ($0). Apologies for the confusion again.

Sign in to join this conversation