huntr

Use After Free in function did_set_string_option in vim/vim

1

✓ Valid Reported on Sep 27th 2022

Description

Use After Free in function did_set_string_option at optionstr.c:2456.

vim version

```
git log
commit 8279af514ca7e5fd3c31cf13b0864163d1a0bfeb (grafted, HEAD -> master, t
```

Proof of Concept

```
./vim -u NONE -i NONE -n -m -X -Z -e -s -S /home/fuzz/test/poc12_huaf.dat -
=================================================================
==57208==ERROR: AddressSanitizer: heap-use-after-free on address 0x62500000
READ of size 8 at 0x625000007518 thread T0
    #0 0x55d956a03ef9 in did_set_string_option /home/fuzz/vim/src/optionstr
    #1 0x55d9569d47cb in do_set_string /home/fuzz/vim/src/option.c:1612
    #2 0x55d9569d7f32 in do_set /home/fuzz/vim/src/option.c:2120
    #3 0x55d9569d2868 in ex_set /home/fuzz/vim/src/option.c:1204
    #4 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_docmd.c:2569
    #5 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990
    #6 0x55d9567d5e84 in do_cmdline_cmd /home/fuzz/vim/src/ex_docmd.c:584
    #7 0x55d956c178ad in f_assert_fails /home/fuzz/vim/src/testing.c:618
    #8 0x55d9567762bb in call_internal_func /home/fuzz/vim/src/evalfunc.c:2
    #9 0x55d956c812d2 in call_func /home/fuzz/vim/src/userfu
    #10 0x55d956c77ae9 in get_func_tv /home/fuzz/vim/src/us
    #11 0x55d956c8cedd in ex_call_inner /home/fuzz/vim/src/userfunc.c:5574
    #12 0x55d956c8ecf0 in ex_call /home/fuzz/vim/src/userfunc.c:5898
    #13 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_docmd.c:2569
    #14 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990

    #15 0x55d956afdb0e in do_source_ext /home/fuzz/vim/src/scriptfile.c:166
    #16 0x55d956afed43 in do_source /home/fuzz/vim/src/scriptfile.c:1811
    #17 0x55d956afb801 in cmd_source /home/fuzz/vim/src/scriptfile.c:1163
    #18 0x55d956afb866 in ex_source /home/fuzz/vim/src/scriptfile.c:1189
    #19 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_docmd.c:2569
    #20 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990
    #21 0x55d9567d5e84 in do_cmdline_cmd /home/fuzz/vim/src/ex_docmd.c:584
    #22 0x55d956de02e8 in exe_commands /home/fuzz/vim/src/main.c:3139
    #23 0x55d956dd945b in vim_main2 /home/fuzz/vim/src/main.c:781
    #24 0x55d956dd8d13 in main /home/fuzz/vim/src/main.c:432
    #25 0x7f2f147b9082 in __libc_start_main ../csu/libc-start.c:308
    #26 0x55d956653e4d in _start (/home/fuzz/vim/src/vim+0x13be4d)

0x625000007518 is located 9240 bytes inside of 9360-byte region [0x62500000
freed by thread T0 here:
    #0 0x7f2f14c5040f in __interceptor_free ../../../../src/libsanitizer/as
    #1 0x55d956654576 in vim_free /home/fuzz/vim/src/alloc.c:623
    #2 0x55d9566691a8 in apply_autocmds_group /home/fuzz/vim/src/autocmd.c:
    #3 0x55d9566673e1 in apply_autocmds /home/fuzz/vim/src/autocmd.c:1709
    #4 0x55d956b3e2d9 in spell_load_lang /home/fuzz/vim/src/spell.c:1600
    #5 0x55d956b40bf5 in did_set_spelllang /home/fuzz/vim/src/spell.c:2105
    #6 0x55d956b4f5a8 in did_set_spell_option /home/fuzz/vim/src/spell.c:44
    #7 0x55d9569ffd6c in did_set_string_option /home/fuzz/vim/src/optionstr
    #8 0x55d9569d47cb in do_set_string /home/fuzz/vim/src/option.c:1612
    #9 0x55d9569d7f32 in do_set /home/fuzz/vim/src/option.c:2120
    #10 0x55d9569d2868 in ex_set /home/fuzz/vim/src/option.c:1204
    #11 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_docmd.c:2569
    #12 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990
    #13 0x55d9567d5e84 in do_cmdline_cmd /home/fuzz/vim/src/ex_docmd.c:584
    #14 0x55d956c178ad in f_assert_fails /home/fuzz/vim/src/testing.c:618
    #15 0x55d9567762bb in call_internal_func /home/fuzz/vim/src/evalfunc.c:
    #16 0x55d956c812d2 in call_func /home/fuzz/vim/src/userfunc.c:3680
    #17 0x55d956c77ae9 in get_func_tv /home/fuzz/vim/src/userfunc.c:1841
    #18 0x55d956c8cedd in ex_call_inner /home/fuzz/vim/src/userfunc.c:5574
    #19 0x55d956c8ecf0 in ex_call /home/fuzz/vim/src/userfun
    #20 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_
    #21 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990
    #22 0x55d956afdb0e in do_source_ext /home/fuzz/vim/src/scriptfile.c:166
    #22 0x55d956afdb0e in do_source_ext /home/fuzz/vim/src/scriptfile.c:16
    #23 0x55d956afed43 in do_source /home/fuzz/vim/src/scriptfile.c:1811
    #24 0x55d956afb801 in cmd_source /home/fuzz/vim/src/scriptfile.c:1163

    #25 0x55d956afb866 in ex_source /home/fuzz/vim/src/scriptfile.c:1189
    #26 0x55d9567e088e in do_one_cmd /home/fuzz/vim/src/ex_docmd.c:2569
    #27 0x55d9567d7aea in do_cmdline /home/fuzz/vim/src/ex_docmd.c:990
    #28 0x55d9567d5e84 in do_cmdline_cmd /home/fuzz/vim/src/ex_docmd.c:584
    #29 0x55d956de02e8 in exe_commands /home/fuzz/vim/src/main.c:3139

previously allocated by thread T0 here:
    #0 0x7f2f14c50808 in __interceptor_malloc ../../../../src/libsanitizer/
    #1 0x55d95665428a in lalloc /home/fuzz/vim/src/alloc.c:246
    #2 0x55d956654120 in alloc_clear /home/fuzz/vim/src/alloc.c:177
    #3 0x55d956679945 in buflist_new /home/fuzz/vim/src/buffer.c:2081
    #4 0x55d956d39928 in win_alloc_firstwin /home/fuzz/vim/src/window.c:387
    #5 0x55d956d3947f in win_alloc_first /home/fuzz/vim/src/window.c:3802
    #6 0x55d956dd97a5 in common_init /home/fuzz/vim/src/main.c:975
    #7 0x55d956dd8a21 in main /home/fuzz/vim/src/main.c:185
    #8 0x7f2f147b9082 in __libc_start_main ../csu/libc-start.c:308

SUMMARY: AddressSanitizer: heap-use-after-free /home/fuzz/vim/src/optionstr
Shadow bytes around the buggy address:
  0x0c4a7fff8e50: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff8e60: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff8e70: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff8e80: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff8e90: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
=>0x0c4a7fff8ea0: fd fd fd[fd]fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff8eb0: fd fd fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4a7fff8ec0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4a7fff8ed0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4a7fff8ee0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4a7fff8ef0: 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 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
  ==57208==ABORTING
```

poc download url: https://github.com/Janette88/vim/blob/main/poc12_huaf.dat

Impact

Referencing memory after it has been freed can cause a program to crash, use unexpected values, or execute code.

CVE
CVE-2022-3352
(Published)

Vulnerability Type
CWE-416: Use After Free

Severity
High (7.8)

Registry
Other

Affected Version
*

Visibility
Public

Status
Fixed

Fixed

Found by

janette88

@janette88

master

Fixed by

Bram Moolenaar

@brammool

maintainer

This report was seen 1,292 times.

We are processing your report and will contact the **vim** team within 24 hours. 2 months ago

We have contacted a member of the **vim** team and are waiting to hear back 2 months ago

Bram Moolenaar validated this vulnerability 2 months ago

I can reproduce it. The POC is usuable for a regression test.

janette88 has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Bram Moolenaar 2 months ago Maintainer

Fixed with patch 9.0.0614

Bram Moolenaar marked this as fixed in **9.0.0614** with commit **ef9763** 2 months ago

Bram Moolenaar has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

Chat with us

Sign in to join this conversation