Heap-based Buffer Overflow in function skip_string in vim/vim

0

✓ Valid Reported on May 7th 2022

Description

Heap-based Buffer Overflow in function skip_string at cindent.c:92

vim version

```
git log
commit 5a8fad32ea9c075f045b37d6c7739891d458f82b (HEAD -> master, tag: v8.2
```

POC

```
./vim -u NONE -i NONE -n -m -X -Z -e -s -S /mnt/share/max/fuzz/poc/vim/poc_
=================================================================
==17385==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x621000
READ of size 1 at 0x621000013d00 thread T0
    #0 0x566e14 in skip_string /home/fuzz/fuzz/vim/vim/src/cindent.c:92:10
    #1 0x5893c2 in find_last_paren /home/fuzz/fuzz/vim/vim/src/cindent.c:1
    #2 0x58dfa6 in cin_isfuncdecl /home/fuzz/fuzz/vim/vim/src/cindent.c:12
    #3 0x580d15 in get_c_indent /home/fuzz/fuzz/vim/vim/src/cindent.c:3835
    #4 0x9999a1 in op_reindent /home/fuzz/fuzz/vim/vim/src/indent.c:1104:1
    #5 0xbab439 in do_pending_operator /home/fuzz/fuzz/vim/vim/src/ops.c:4
    #6 0xb1a7a5 in normal_cmd /home/fuzz/fuzz/vim/vim/src/normal.c:952:2
    #7 0x80ebde in exec_normal /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:8757
    #8 0x80e408 in exec_normal_cmd /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:
    #9 0x80dfb9 in ex_normal /home/fuzz/fuzz/vim/vim/src/ex
    #10 0x7d7529 in do_one_cmd /home/fuzz/fuzz/vim/vim/src/
    #11 0x7c42e5 in do_cmdline /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:992:
```

```
    #12 0xe5191c in do_source_ext /home/fuzz/fuzz/vim/vim/src/scriptfile.c:
    #13 0xe4e376 in do_source /home/fuzz/fuzz/vim/vim/src/scriptfile.c:1801
    #14 0xe4dcac in cmd_source /home/fuzz/fuzz/vim/vim/src/scriptfile.c:117

    #15 0xe4d38e in ex_source /home/fuzz/fuzz/vim/vim/src/scriptfile.c:120
    #16 0x7d7529 in do_one_cmd /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:2567:
    #17 0x7c42e5 in do_cmdline /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:992:1
    #18 0x7c8f31 in do_cmdline_cmd /home/fuzz/fuzz/vim/vim/src/ex_docmd.c:5
    #19 0x1419502 in exe_commands /home/fuzz/fuzz/vim/vim/src/main.c:3108:2
    #20 0x141569b in vim_main2 /home/fuzz/fuzz/vim/vim/src/main.c:780:2
    #21 0x140ad95 in main /home/fuzz/fuzz/vim/vim/src/main.c:432:12
    #22 0x7fe4d382b082 in __libc_start_main /build/glibc-SzIz7B/glibc-2.31/
    #23 0x41ea6d in _start (/home/fuzz/fuzz/vim/vim/src/vim+0x41ea6d)

0x621000013d00 is located 0 bytes to the right of 4096-byte region [0x6210
allocated by thread T0 here:
    #0 0x499ccd in malloc (/home/fuzz/fuzz/vim/vim/src/vim+0x499ccd)
    #1 0x4cb3aa in lalloc /home/fuzz/fuzz/vim/vim/src/alloc.c:246:11
    #2 0x4cb28a in alloc /home/fuzz/fuzz/vim/vim/src/alloc.c:151:12
    #3 0x1422fb5 in mf_alloc_bhdr /home/fuzz/fuzz/vim/vim/src/memfile.c:884
    #4 0x1421dc7 in mf_new /home/fuzz/fuzz/vim/vim/src/memfile.c:375:26
    #5 0xa5be28 in ml_new_data /home/fuzz/fuzz/vim/vim/src/memline.c:4082:1
    #6 0xa5a7d1 in ml_open /home/fuzz/fuzz/vim/vim/src/memline.c:394:15
    #7 0x4fddba in open_buffer /home/fuzz/fuzz/vim/vim/src/buffer.c:186:9
    #8 0x1416d4c in create_windows /home/fuzz/fuzz/vim/vim/src/main.c:2877:
    #9 0x141501a in vim_main2 /home/fuzz/fuzz/vim/vim/src/main.c:711:5
    #10 0x140ad95 in main /home/fuzz/fuzz/vim/vim/src/main.c:432:12
    #11 0x7fe4d382b082 in __libc_start_main /build/glibc-SzIz7B/glibc-2.31/

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/fuzz/fuzz/vim/vim/src
Shadow bytes around the buggy address:
  0x0c427fffa750: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c427fffa760: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c427fffa770: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c427fffa780: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c427fffa790: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c427fffa7a0:[fa]fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7b0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7c0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7d0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7e0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7f0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c427fffa7f0: 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
==17385==ABORTING
```

poc_h7_s.dat

Impact

This vulnerabilities are capable of crashing software, Modify Memory, and possible remote execution

Occurrences

C cindent.c L92

CVE
CVE-2022-1733
(Published)

Chat with us

Vulnerability Type
CWE-122: Heap-based Buffer Overflow

Severity

Medium (6.6)

Registry
Other

Affected Version
*

Visibility
Public

Status
Fixed

Found by

TDHX ICS Security
@jieyongma
pro

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 1,045 times.

We are processing your report and will contact the **vim** team within 24 hours. 7 months ago

We have contacted a member of the **vim** team and are waiting to hear back 7 months ago

Bram Moolenaar 7 months ago Maintainer

I cannot reproduce it with valgrind (ASAN requires rebuilding).
The POC looks complicated, the stack suggests the only thing needed is to get the buffer content text in a certain state before executing the tilde operator.

We have sent a follow up to the **vim** team. We will try again in 7 days. 6 months ago

TDHX ICS Security modified the report 6 months ago

TDHX 6 months ago Researcher

I cannot reproduce the orignal issue either, but found another location with the same issue, so the report is updated to the new location with new poc file, hope you can reproduce it.

Bram Moolenaar 6 months ago Maintainer

Yes, with this POC I can reproduce the problem. And the POC is simple enough to use for a test. I'll fix it.

Bram Moolenaar validated this vulnerability 6 months ago

TDHX ICS Security has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Bram Moolenaar 6 months ago Maintainer

Fixed with v8.2.4968

Bram Moolenaar marked this as fixed in 8.2 with commit 60ae0e 6 months ago

Bram Moolenaar has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

cindent.c#L92 has been validated ✓

Sign in to join this conversation

Chat with us