Heap-based buffer overflow in function ins_compl_add in vim/vim

0

Valid Reported on Jul 6th 2022

Description

Heap-based buffer overflow in function `ins_compl_add` at insexpand.c:751

Version

```
commit b8329db36a886355e6e9cb9986a3668fef78c438 (HEAD -> master, tag: v9.0.
```

Proof of Concept

```
guest@elk:~/trung$ valgrind ./vim_latest/src/vim  -u NONE -i NONE -n -m -X
==2961== Memcheck, a memory error detector
==2961== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==2961== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright infc
==2961== Command: ./vim_latest/src/vim -u NONE -i NONE -n -m -X -Z -e -s -S
==2961==
==2961== Invalid read of size 1
==2961==    at 0x1FAA73: ins_compl_add (insexpand.c:751)
==2961==    by 0x1FAEE9: ins_compl_add_infercase (insexpand.c:697)
==2961==    by 0x2B5879: find_pattern_in_path (search.c:3768)
==2961==    by 0x1FDAAA: get_next_include_file_completion (insexpand.c:3307
==2961==    by 0x1FDAAA: get_next_completion_match (insexpand.c:3659)
==2961==    by 0x1FDAAA: ins_compl_get_exp (insexpand.c:3767)
==2961==    by 0x1FDAAA: find_next_completion_match (insexpand.c:4002)
==2961==    by 0x1FDAAA: ins_compl_next (insexpand.c:4103)
==2961==    by 0x1FEA8A: ins_complete (insexpand.c:4954)
==2961==    by 0x180846: edit (edit.c:1281)
```

```
==2961==    by 0x22FB09: invoke_edit.isra.1 (normal.c:7037)
==2961==    by 0x231D41: n_opencmd (normal.c:6281)
==2961==    by 0x231D41: nv_open (normal.c:7418)

==2961==    by 0x238B24: normal_cmd (normal.c:939)
==2961==    by 0x1B6ADC: exec_normal (ex_docmd.c:8809)
==2961==    by 0x1B6D3F: ex_normal (ex_docmd.c:8695)
==2961==    by 0x1BB65D: do_one_cmd (ex_docmd.c:2570)
==2961==    by 0x1BB65D: do_cmdline (ex_docmd.c:992)
==2961==  Address 0x608788a is 2 bytes after a block of size 8 alloc'd
==2961==    at 0x4C31B0F: malloc (in /usr/lib/valgrind/vgpreload_memcheck-a
==2961==    by 0x140E20: lalloc (alloc.c:246)
==2961==    by 0x2DAFA4: vim_strnsave (strings.c:44)
==2961==    by 0x1FAACD: ins_compl_add (insexpand.c:768)
==2961==    by 0x1FAEE9: ins_compl_add_infercase (insexpand.c:697)
==2961==    by 0x1FCF18: get_next_default_completion (insexpand.c:3629)
==2961==    by 0x1FCF18: get_next_completion_match (insexpand.c:3694)
==2961==    by 0x1FCF18: ins_compl_get_exp (insexpand.c:3767)
==2961==    by 0x1FCF18: find_next_completion_match (insexpand.c:4002)
==2961==    by 0x1FCF18: ins_compl_next (insexpand.c:4103)
==2961==    by 0x1FEA8A: ins_complete (insexpand.c:4954)
==2961==    by 0x180846: edit (edit.c:1281)
==2961==    by 0x22FB09: invoke_edit.isra.1 (normal.c:7037)
==2961==    by 0x231D41: n_opencmd (normal.c:6281)
==2961==    by 0x231D41: nv_open (normal.c:7418)
==2961==    by 0x238B24: normal_cmd (normal.c:939)
==2961==    by 0x1B6ADC: exec_normal (ex_docmd.c:8809)
==2961==
==2961==
==2961== HEAP SUMMARY:
==2961==     in use at exit: 73,848 bytes in 396 blocks
==2961==   total heap usage: 1,852 allocs, 1,456 frees, 3,028,954 bytes all
==2961==
==2961== LEAK SUMMARY:
==2961==    definitely lost: 0 bytes in 0 blocks
==2961==    indirectly lost: 0 bytes in 0 blocks
==2961==      possibly lost: 148 bytes in 8 blocks
==2961==    still reachable: 73,700 bytes in 388 blocks
==2961==         suppressed: 0 bytes in 0 blocks
==2961== Rerun with --leak-check=full to see details of lea
==2961==
```

```
==2961== For counts of detected and suppressed errors, rerun with: -v
==2961== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

Attachment

poc42min

Impact

This may result in corruption of sensitive information, a crash, or code execution among other things.

CVE
CVE-2022-2344
(Published)

Vulnerability Type
CWE-122: Heap-based Buffer Overflow

Severity
High (7.8)

Registry
Other

Affected Version
9.0.0044

Visibility
Public

Status
Fixed

Found by

xikhud
@acquykhud
legend

Fixed by

Bram Moolenaar
@brammool

maintainer

This report was seen 819 times.

We are processing your report and will contact the **vim** team within 24 hours. 5 months ago

We have contacted a member of the **vim** team and are waiting to hear back 5 months ago

Bram Moolenaar validated this vulnerability 5 months ago

I can reproduce it. The POC looks simple enough to use for a test.

xikhud 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 **9.0.0045** with commit **baefde** 5 months ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Sign in to join this conversation