Heap-based Buffer Overflow in vim/vim

0

Valid Reported on Jan 8th 2022

Description

Heap-buffer-overflow in vim

Command

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

Proof of Concept

minpoc is here. #bt

```
Program received signal SIGABRT, Aborted.
__GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:50
50      ../sysdeps/unix/sysv/linux/raise.c: No such file or directory.
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA
─────────────────────────────────────────────────────────────────────────
 RAX  0x0
 RBX  0x7ffff771c880 ◂— 0x7ffff771c880
 RCX  0x7ffff787718b (raise+203) ◂— mov    rax, qword ptr [rsp + 0x108]
 RDX  0x0
 RDI  0x2
 RSI  0x7fffffffbee0 ◂— 0x0
 R8   0x0
 R9   0x7fffffffbee0 ◂— 0x0
 R10  0x8
 R11  0x246
 R12  0x7fffffffc150 —▸ 0x805870 ◂— 0x7f00367074 /* 'tp6' */
 R13  0x10
 R14  0x7ffff7ffb000 ◂— 0x6c6c616d00001000
 R15  0x1
 RBP  0x7fffffffc230 ◂— 0x2de
```

```
RSP  0x7fffffffbee0 ◂— 0x0
RIP  0x7ffff787718b (raise+203) ◂— mov    rax, qword ptr [rsp + 0x108]
```

```
► 0x7ffff787718b <raise+203>    mov    rax, qword ptr [rsp + 0x108]
  0x7ffff7877193 <raise+211>    xor    rax, qword ptr fs:[0x28]
  0x7ffff787719c <raise+220>    jne    raise+260                <raise+260
   ↓
  0x7ffff78771c4 <raise+260>    call   __stack_chk_fail                <__

  0x7ffff78771c9                nop    dword ptr [rax]
  0x7ffff78771d0 <killpg>       endbr64
  0x7ffff78771d4 <killpg+4>     test   edi, edi
  0x7ffff78771d6 <killpg+6>     js     killpg+16                <killpg+16

  0x7ffff78771d8 <killpg+8>     neg    edi
  0x7ffff78771da <killpg+10>    jmp    kill                <kill>

  0x7ffff78771df <killpg+15>    nop
```

```
00:0000│ rsi r9 rsp 0x7fffffffbee0 ◂— 0x0
01:0008│            0x7fffffffbee8 —▸ 0x663998 (check_termcode+72) ◂— cmp
02:0010│            0x7fffffffbef0 ◂— 0x40 /* '@' */
03:0018│            0x7fffffffbef8 ◂— 0x7000000101
04:0020│            0x7fffffffbf00 ◂— 0x8
05:0028│            0x7fffffffbf08 ◂— 0x1
06:0030│            0x7fffffffbf10 ◂— 0x0
07:0038│            0x7fffffffbf18 ◂— 0x770000007c /* '|' */
```

```
► f 0   0x7ffff787718b raise+203
  f 1   0x7ffff7856859 abort+299
  f 2   0x7ffff78c13ee __libc_message+670
  f 3   0x7ffff78c947c
  f 4   0x7ffff78cc83a _int_malloc+3146
  f 5   0x7ffff78ce2d4 malloc+116
  f 6         0x4063a7 lalloc+87
  f 7         0x40634a alloc+26
```

```
pwndbg>
pwndbg> bt
#0  __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:50
```

```
#1  0x00007ffff7856859 in __GI_abort () at abort.c:79
#2  0x00007ffff78c13ee in __libc_message (action=action@entry=do_abort, fmt
#3  0x00007ffff78c947c in malloc_printerr (str=str@entry=0x7ffff79e9556 "ma

#4  0x00007ffff78cc83a in _int_malloc (av=av@entry=0x7ffff7a1cb80 <main_are
#5  0x00007ffff78ce2d4 in __GI___libc_malloc (bytes=734) at malloc.c:3058
#6  0x00000000004063a7 in lalloc (size=734, message=1) at alloc.c:248
#7  0x000000000040634a in alloc (size=734) at alloc.c:151
#8  0x0000000000561418 in block_insert (oap=0x7fffffffc7c8, s=0x8d0dc0 "HI4
#9  0x000000000056116e in op_insert (oap=0x7fffffffc7c8, count1=1) at ops.c
#10 0x00000000005663cc in do_pending_operator (cap=0x7fffffffc758, old_col=
#11 0x000000000054f207 in normal_cmd (oap=0x7fffffffc7c8, toplevel=1) at nc
#12 0x00000000004aa97a in exec_normal (was_typed=0, use_vpeekc=0, may_use_t
#13 0x00000000004aa81b in exec_normal_cmd (cmd=0x805355 "0r\te\026\067QG4QA
#14 0x00000000004aa73c in ex_normal (eap=0x7fffffffcb38) at ex_docmd.c:8516
#15 0x00000000004a1535 in do_one_cmd (cmdlinep=0x7fffffffd3d8, flags=7, cst
#16 0x000000000049e6e2 in do_cmdline (cmdline=0x805220 "00", fgetline=0x5fe
#17 0x00000000005fe817 in do_source (fname=0x7fd963 "/home/zxq/CVE_testing/
#18 0x00000000005fdbc6 in cmd_source (fname=0x7fd963 "/home/zxq/CVE_testing
#19 0x00000000005fdadc in ex_source (eap=0x7fffffffd798) at scriptfile.c:10
#20 0x00000000004a1535 in do_one_cmd (cmdlinep=0x7fffffffe038, flags=11, cs
#21 0x000000000049e6e2 in do_cmdline (cmdline=0x7fd900 "so /home/zxq/CVE_te
#22 0x000000000049f334 in do_cmdline_cmd (cmd=0x7fd900 "so /home/zxq/CVE_te
#23 0x0000000000728903 in exe_commands (parmp=0x7e8a58 <params>) at main.c:
#24 0x000000000072795a in vim_main2 () at main.c:774
#25 0x00000000007252c1 in main (argc=11, argv=0x7fffffffe238) at main.c:426
#26 0x00007ffff78580b3 in __libc_start_main (main=0x724d60 <main>, argc=11,
#27 0x000000000040617e in _start ()
```

CVE
CVE-2022-0261
(Published)

Vulnerability Type
CWE-122: Heap-based Buffer Overflow

Severity
None (0)

Visibility
Public

Status
Fixed

Found by

zfeixq
@zfeixq
unranked

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 1,098 times.

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

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

Bram Moolenaar a year ago — Maintainer

This POC is much too long. Please reduce it to the minimal needed to reproduce the issue.

We have sent a follow up to the **vim** team. We will try again in 7 days. 10 months ago

zfeixq 10 months ago — Researcher

POC is here.

zfeixq modified the report 10 months ago

zfeixq modified the report 10 months ago

Bram Moolenaar validated this vulnerability 10 months ago

zfeixq has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

Bram Moolenaar 10 months ago Maintainer

Thanks for the new POC, I can reproduce the problem. I'll make a fix.

Bram Moolenaar 10 months ago Maintainer

Found a simpler way to reproduce the problem. With the fix it is in patch 8.2.4120.

Bram Moolenaar marked this as fixed in 8.2 with commit 9f8c30 10 months ago

Bram Moolenaar has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Sign in to join this conversation