

New issue

Jump to bottom

## Second segfault in lj\_err\_run #603

 Closed Changochen opened this issue on Jul 13, 2020 · 3 comments

Labels 2.0 2.1 bug

Changochen commented on Jul 13, 2020 • edited

Hi, we found a crash in LuaJIT

Version: 2.1. Git hash: 570e758ca7dd14f93efdd43d68cf8979c1d7f984


POC:

```
function errfunc() print(xpcall(test, errfunc)) end function test(do_yield) mt =
{} local t print(xpcall(test, errfunc)) function errfunc() end function
test() end coroutine.wrap(function() end) coro() end setmetatable(xpcall(
test,
function()
print(coroutine.resume(coroutine.create(function() coroutine.resume(
coroutine.create(function() collectgarbage() end)) end))) end))
```

Stack dump:

```
AddressSanitizer:DEADLYSIGNAL
=====
==4659==ERROR: AddressSanitizer: SEGV on unknown address 0x7f5b27b05000 (pc 0x0000004c779e bp 0x7ffd4ac1c930 sp 0x7ffd4ac1c840 T0)
==4659==The signal is caused by a READ memory access.
#0 0x4c779d in lj_err_run /home/yongheng/LuaJit_asan/src/lj_err.c:608:10
#1 0x4c821e in lj_err_callermsg /home/yongheng/LuaJit_asan/src/lj_err.c:724:3
#2 0x4c877f in err_argmsg /home/yongheng/LuaJit_asan/src/lj_err.c:756:3
#3 0x4c8886 in lj_err_argtype /home/yongheng/LuaJit_asan/src/lj_err.c:796:3
#4 0x4c89dd in lj_err_argt /home/yongheng/LuaJit_asan/src/lj_err.c:802:3
#5 0x60ec7a in lj_l1b_checktab /home/yongheng/LuaJit_asan/src/lj_l1b.c:270:5
#6 0x60f845 in lj_ffh_setmetatable /home/yongheng/LuaJit_asan/src/l1b_base.c:131:14
#7 0x54a45a in lj_fff_fallback (/home/yongheng/LuaJit_asan/src/luajit+0x54a45a)
#8 0x4c78f1 in lj_err_run /home/yongheng/LuaJit_asan/src/lj_err.c:617:5
#9 0x4c7b54 in err_msgv /home/yongheng/LuaJit_asan/src/lj_err.c:632:3
#10 0x4c7928 in lj_err_msg /home/yongheng/LuaJit_asan/src/lj_err.c:638:3
#11 0x4d3799 in lj_state_growstack /home/yongheng/LuaJit_asan/src/lj_state.c:118:5
#12 0x5797e3 in lj_snap_restore /home/yongheng/LuaJit_asan/src/lj_snap.c:874:5
#13 0x52a685 in trace_exit_cp /home/yongheng/LuaJit_asan/src/lj_trace.c:793:13
#14 0x548bca in lj_vm_cpcall (/home/yongheng/LuaJit_asan/src/luajit+0x548bca)
#15 0x529161 in lj_trace_exit /home/yongheng/LuaJit_asan/src/lj_trace.c:863:13
#16 0x54a7d1 in lj_vm_exit_handler (/home/yongheng/LuaJit_asan/src/luajit+0x54a7d1)
#17 0x548555 in lj_BC_IFORL (/home/yongheng/LuaJit_asan/src/luajit+0x548555)
#18 0x7f5b26bb11af (<unknown module>)


AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/yongheng/LuaJit_asan/src/lj_err.c:608:10 in lj_err_run
==4659==ABORTING
```

 MikePall added 2.0 2.1 bug labels on Aug 9, 2020

MikePall commented on Aug 9, 2020

Member

Fixed. Thanks!

 MikePall closed this as completed on Aug 9, 2020

zsh-igtm commented on Aug 17, 2020

Hey, would you mind giving a link to the change that fixed this?

 2

fsfod commented on Aug 17, 2020 • edited

It probably these two commits "Fix handling of errors during snapshot restore" and "Call error function on rethrow after trace exit", since the crashing call stack is throwing an error when failing to grow the Lua stack inside snapshot restore during a trace exit.

 3

 vcunat added a commit to NixOS/nixpkgs that referenced this issue on Aug 31, 2020

 lua-jit\*: update to address CVE-2020-24372

0e58393

 This was referenced on Sep 29, 2020

Merge LuaJIT/LuaJIT/v2.1 with v2.1-agentzh openresty/luajit2#104

 Merged

Adjust stackov.lua to expect "error in error handling" openresty/luajit2-test-suite#8

 Merged

Assignees

No one assigned

Labels

2.0 2.1 bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants

