New issue

Jump to bottom

SIGABRT in jerry_port_fatal #3860

Closed ArayzWang opened this issue on Jun 4, 2020 · 3 comments · Fixed by #3867

Labels bug ecma core

ArayzWang commented on Jun 4, 2020 • edited

JerryScript revision

`c09c2c5`

Build platform

Ubuntu 18.04 LTS

Build steps

python tools/build.py --profile=es2015-subset --lto=off --error-messages=on --strip=off --compile-flag=-fsanitize=address

Test case

```
function main() {
const v4 = [13.37,13.37];
const v6 = [1337,1337];
const v7 = [2147483649,13.37,"species"];
const v8 = {a:13.37,length:13.37};
const v9 = {constructor:v7};
let v10 = v6;
const v12 = {};
const v13 = [v12,v12,v12,v12];
const v14 = gc();
const v15 = "species".__proto__;
let v18 = 0;
while (v18 < 4) {
    const v19 = v18 + 1;
    v18 = v19;
}
const v22 = gc();
const v23 = "species".length;
const v24 = [13.37,13.37];
const v26 = [1337,1337];
const v27 = [2147483649,13.37,"species"];
const v28 = {a:13.37,length:13.37};
const v29 = {constructor:v27};
const v33 = [13.37,13.37];
const v36 = {get:gc,set:gc};
const v38 = Object.defineProperty(v9,"e",v36);
const v39 = !v8;
const v40 = [0,v33,"species"];
const v45 = [2147483649,13.37,"species"];
const v46 = {constructor:v45};
const v48 = [1337,1337];
const v51 = Object();
let v52 = 0;
const v53 = Object();
const v54 = v52 + 1;
const v55 = [1337,1337];
const v56 = [2147483649,1337,"species"];
const v57 = {a:13.37,length:13.37};
const v58 = {constructor:v56};
const v61 = [1337,1337];
let v62 = v61;
const v66 = [1337,1337];
const v67 = [2147483649,"species","species"];
const v69 = v66 % v66;
const v70 = [1337,1337];
const v71 = !v46;
let v74 = 0;
do {
    let v75 = 13.37;
    try {
        const v76 = Object(...1,v38);
    } catch(v77) {
        const v78 = typeof v56;
        const v80 = v78 === "number";
        let v81 = v80;
    }
} while (v74 < 9);
}
main();
```

Execution steps

build/bin/jerry testcase.js

Output

Program received signal SIGABRT, Aborted.

Backtrace

Program received signal SIGABRT, Aborted.
__GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:51
(gdb) bt

```
#0  __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:51
#1  0x00007ffff6e43801 in __GI_abort () at abort.c:79
#2  0x00000000005ac952 in jerry_port_fatal ()
#3  0x000000000053beef in jerry_fatal ()
#4  0x00000000004f90d3 in ecma_ref_object ()
#5  0x00000000005042cd in ecma_copy_value ()
#6  0x000000000055cf7c in vm_loop ()
#7  0x000000000055b5f6 in vm_execute ()
#8  0x000000000055b193 in vm_run ()
#9  0x000000000051f650 in ecma_op_function_call_simple ()
#10 0x000000000051f2d6 in ecma_op_function_call ()
#11 0x000000000055b9aa in vm_execute ()
#12 0x000000000055b193 in vm_run ()
#13 0x00000000004f501e in jerry_run ()
#14 0x00000000004f25df in main ()
```

akosthekiss commented on Jun 5, 2020 Member

Thanks for the report. One comment though. "SIGABRT in jerry_port_fatal" is not the best way to describe this issue as `jerry_port_fatal` and `jerry_fatal` are simply the internal error handling routines of the engine. That is, the error is actually reported by `ecma_ref_object`. And SIGABRT is only how `jerry_port_fatal` terminates the engine in case of an internal error. To get more descriptive error messages, you should add `--logging=on` to your build options.

akosthekiss commented on Jun 5, 2020 Member

Plus, if you put the backtrace in a fenced code block (triple backticks) then the stack frames (e.g., `#14`) will not be incorrectly linked to past issues or PRs.

The same goes to the test case. Guarding it as a code block will prevent the Markdown engine to re-style `__proto__` as **proto**.

ArayzWang commented on Jun 5, 2020 Author

> Plus, if you put the backtrace in a fenced code block (triple backticks) then the stack frames (e.g., `#14`) will not be incorrectly linked to past issues or PRs.
>
> The same goes to the test case. Guarding it as a code block will prevent the Markdown engine to re-style `__proto__` as **proto**.

Thanks for the suggestion.

dbatyai added **bug** **ecma core** labels on Jun 5, 2020

galpeter mentioned this issue on Jun 5, 2020

Correct release of spread arguments #3867

Merged

zherczeg closed this as completed in #3867 on Jun 6, 2020

Assignees

No one assigned

Labels

bug **ecma core**

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

Correct release of spread arguments
galpeter/jerryscript

3 participants