

Copy Summary

View

Closed

Bug 1715318 (CVE-2021-29982)

Opened 2 years ago

Closed 2 years ago

Differential Testing: Different output during TypeError

Categories

Product: Core

Type: defect

Component: JavaScript Engine: JIT

Priority: P1

Severity: S3

Tracking

Status: RESOLVED FIXED

Tracking Flags: firefox91

Tracking ---

Status fixed

Milestone: 91 Branch

People

(Reporter: lukas.bernhard, Assigned: jandem)

References

(Blocks 1 open bug)

Details

(Keywords: csetype-disclosure, sec-low, Whiteboard: [adv-main91+])

Attachments

Bug 1715318 part 1 - Fix LCheckClassHeritage to not clobber the object register. r?iain!

2 years ago Jan de Mooij [jandem]

48 bytes, text/x-phabricator-request

Details | Review

Bug 1715318 part 2 - Use fallibleUnboxObject for LCheckClassHeritage. r?iain!

2 years ago Jan de Mooij [jandem]

48 bytes, text/x-phabricator-request

Details | Review

advisory.txt

1 year ago Tom Ritter [tjr]

249 bytes, text/plain

Details

Bottom

Tags

Timeline

lukas.bernhard

Reporter

Description • 2 years ago

—

User Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0

Steps to reproduce:

The following testcase produces different results on mozilla-central git commit 0e97d943bc746e4fd92902b21d3d5bebfef885a1 depending on whether ion is enabled or disabled.

```
function main() {
  let v1 = 0;
  function v18() {
    for (let v35 = 0; v35 < 100; v35++) {}

    try {
      async function* v37() {}
      class V47 extends v37 {};
      v1 += 1;
    }
    catch (e) {
      print("caught: " + e);
    }
  };
  v18();
  v18(); // second call may or may not reach catch handler, approx 50/50
  print(v1); // either 0 or 1
}
main();
```

Actual results:

Running the sample with ion disabled (obj-x86_64-pc-linux-gnu/dist/bin/js --no-threads --cpu-count=1 --baseline-warmup-threshold=10 --fuzzing-safe --differential-testing --no-ion diff.js) causes two exceptions to be thrown, v1 is 0 at the end of main().

If ion is enabled (obj-x86_64-pc-linux-gnu/dist/bin/js --no-threads --cpu-count=1 --ion-offthread-compile=off --baseline-warmup-threshold=10 --ion-warmup-threshold=100 --ion-check-range-analysis --ion-extra-checks --fuzzing-safe --differential-testing diff.js), the second execution of v18() may or may not (approx. 50/50) throw an exception. v1 is either 0 or 1 at the end of main().

Marking as security as a few differential execution bugs were flagged in the past.

Christian Holler (decoder)

Updated • 2 years ago

—

Group: firefox-core-security → core-security

Component: Untriaged → JavaScript Engine: JIT

Product: Firefox → Core

Version: Firefox 91 → Trunk

J

Jan de Mooij [jandem]

Assignee

Updated • 2 years ago

—

Flags: needinfo?(jdemooij)

Daniel Veditz [dveditz]

Updated • 2 years ago

—

Group: core-security → javascript-core-security

J

Jan de Mooij [jandem]

Assignee

Comment 1 • 2 years ago

—

J

Jan de Mooij [jandem]

Updated • 2 years ago

Assignee

Assignee: nobody → jdemooij
Status: UNCONFIRMED → ASSIGNED
Ever confirmed: true

Christian Holler (:decoder)

Updated • 2 years ago

Flags: sec-bounty?

J

Jan de Mooij [jandem]

Comment 2 • 2 years ago

Assignee

Opening this up. We end up reading a single bit from one of the pointers in the function `JSC::Class` and interpreting that as "is constructor", so ASLR causes the differential behavior, but this isn't exploitable.

Jan de Mooij [jandem]

Comment 3 • 2 years ago

Assignee

Attached file [Bug 1715318 part 1 - Fix LCheckClassHeritage to not clobber the object register. r?iain! — Details](#)

isCallableOrConstructor first does a JSC::Class check which clobbers the output register. If the JSC::Class is the JSC::Function class, it then checks the function's is-constructor flag. If we pass the same register for object and output, we end up reading a garbage bit of the JSC::Function JSC::Class and interpreting that as is-constructor.

J

Jan de Mooij [jandem]

Comment 4 • 2 years ago

Assignee

Attached file [Bug 1715318 part 2 - Use fallibleUnboxObject for LCheckClassHeritage. r?iain! — Details](#)

This is a bit simpler and faster.

Depends on D117634

J

Jan de Mooij [jandem]

Updated • 2 years ago

Assignee

Flags: ~~needsinfo(jdemooij)~~

Pulsebot

Comment 5 • 2 years ago

Pushed by jdemooij@mozilla.com:
<https://hg.mozilla.org/integration/autoland/rev/7f9481bf5840>
part 1 - Fix LCheckClassHeritage to not clobber the object register. r=iain
<https://hg.mozilla.org/integration/autoland/rev/09af51138cf0>
part 2 - Use fallibleUnboxObject for LCheckClassHeritage. r=iain

Matthew Gaudet (he/him) [:mgaudet]

Updated • 2 years ago

Severity: -- → S3
Priority: -- → P1

Sandor Molnar

Comment 6 • 2 years ago

bugherder

Jan de Mooij [jandem]

Updated • 2 years ago

Assignee

<https://hg.mozilla.org/mozilla-central/rev/7f9481bf5840>
<https://hg.mozilla.org/mozilla-central/rev/09af51138cf0>

Status: ASSIGNED → RESOLVED
Closed: 2 years ago
[status-firefox91](#): --- → fixed
Resolution: --- → FIXED
Target Milestone: --- → 91 Branch

Daniel Veditz [:dveditz]

Comment 7 • 2 years ago

A one-bit leak is still a leak so we'll tag this as a low-severity security bug.

Jan de Mooij [jandem]

Updated • 2 years ago

Assignee


Flags: sec-bounty? → sec-bounty+
Keywords: [csectype-disclosure](#), [sec-low](#)

Tom Ritter [:tjr]

Updated • 1 year ago

Assignee

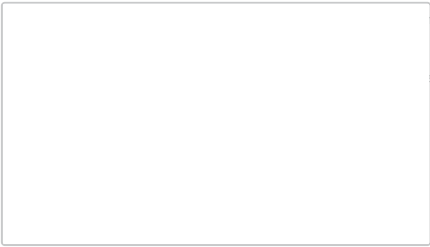
Whiteboard: [adv-main90+]

**Tom Ritter [tjr]**
Updated • 1 year ago

Whiteboard: [adv-main90+] → [adv-main91+]

**Tom Ritter [tjr]**
Comment 8 • 1 year ago

Attached file [advisory.txt](#) — [Details](#)



**Tom Ritter [tjr]**
Updated • 1 year ago

Alias: CVE-2021-29982

**Nicolas B. Pierron [nbp]** [away until 2023-01-02]
Updated • 9 months ago

Blocks: [t11d-js-fuzzing](#)

You need to [log in](#) before you can comment on or make changes to this bug.

Top ↑