

ClosedBug 1640737 (CVE-2020-12417)Opened 3 years agoClosed 3 years ago

Assertion failure: LoadElement instruction returned value with unexpected type, at js/src/jit/MacroAssembler.cpp:1862

Categories

Product: Core▼
Component: JavaScript Engine: JIT▼
Platform: ARM64Linux

Type: defect
Priority: P1Severity: S2

Tracking

Status: RESOLVED FIXED
Milestone: mozilla78

Tracking Flags:
firefox-esr68
firefox-esr78
firefox76
firefox77
firefox78

Tracking
78+ fixed
78+ fixed
--- wontfix
- wontfix
+ fixed

Status
fixed
fixed
wontfix
wontfix
fixed

People

(Reporter: deian, Assigned: jandem)

Details

(Keywords: sec-high, Whiteboard: [post-critsmash-triage][sec-survey][adv-main78+][adv-esr68.10+])

Attachments

Bug 1640737 part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iaim!3 years agoJan de Mooij [jandem]47 bytes, text/x-phabricator-request

dveditz : sec-approval+Details | Review

Bug 1640737 part 2 - Add a test, remove now unused code. r?iaim!3 years agoJan de Mooij [jandem]47 bytes, text/x-phabricator-request

Details | Review

Patch for ESR683 years agoJan de Mooij [jandem]6.41 KB, patch

RyanVM : approval-mozilla-esr68+Details | Diff | Splinter Review

advisory.txt3 years agoTom Ritter [tjr]297 bytes, text/plain

Details

Bottom ▼Tags ▼Timeline ▼

Deian StefanReporterDescription • 3 years ago


—

Our JITing produced (run with --ion-eager --ion-offthread-compile=off) two crash files, both triggering the assert [here](#). Unfortunately, the stack trace wasn't very useful, so I'm not sure what the root cause is. (On x86 this doesn't crash.)

```
function main() {
  let v2 = 0;
  do {
    const v3 = v2 + 1;
    v2 = v3;
  } while (v2 < 7);
  for (let v7 = 0; v7 < 2; v7++) {
  }
  let v10 = 0;
  while (v10 < 8) {
    for (let v14 = 0; v14 < 6; v14++) {
    }
    const v16 = [13.37,13.37,13.37];
    const v18 = [1337,1337];
    const v19 = {__proto__:v18,e:1337,valueOf:v16};
    for (let v23 = 0; v23 < 100; v23++) {
      const v25 = [13.37,v19];
      const v26 = v25.flat();
    }
  }
}
main();
gc();

function main() {
  let v2 = 0;
  do {
    const v3 = v2 + 1;
    v2 = v3;
  } while (v2 < 7);
  for (let v7 = 0; v7 < 2; v7++) {
  }
  let v10 = 0;
  while (v10 < 8) {
    for (let v14 = 0; v14 < 6; v14++) {
    }
    const v16 = [13.37,13.37,13.37];
    const v18 = [1337,1337];
    const v19 = {__proto__:v18,e:1337,valueOf:v16};
    for (let v23 = 0; v23 < 100; v23++) {
      const v25 = [13.37,v19];
      const v26 = v25.flat();
    }
  }
}
}
```


```
}
main();
gc();
```

**Daniel Veditz** [dveditz]

Updated • 3 years ago

—

Group: core-security → javascript-core-security


**Jan de Mooij** [jandem]

Comment 1 • 3 years ago

Assignee

—

The two tests are identical AFAICS?


**Deian Stefan**

Comment 2 • 3 years ago

Reporter

—

Yep, sorry! I also double checked the files to make sure I didn't copy incorrectly. Apparently the fuzzer thought they were different cases. Woops.

**Jan de Mooij** [jandem]


Comment 3 • 3 years ago

Assignee

—

No problem, thanks for reporting! The not-x86 thing is interesting.
More minimal test that doesn't rely on self-hosted code:

```
function g(arr) {
  var res = [];
  for (var i = 0; i < arr.length; i++) {
    var el = arr[i];
    res.push(el);
  }
  return res;
}
function f() {
  for (var i = 0; i < 2; i++) {
    var obj = {__proto__: []};
    for (var j = 0; j < 100; j++) {
      g([13.37, obj]);
    }
  }
}
f();
```

**Jan de Mooij** [jandem]


Comment 4 • 3 years ago

Assignee

—

I think I know what's going on here. Really good find.

Assignee: nobody → jdemooij
Status: NEW → ASSIGNED

**Jan de Mooij** [jandem]

Comment 5 • 3 years ago

Assignee

—

This is an ARM64-specific bug. Exploitable, could be causing crashes in the wild.
The culprit is the MacroAssembler::extractTag implementation for TypedOrValueRegister :

```
MOZ_MUST_USE Register extractTag(const TypedOrValueRegister& reg,
                                Register scratch) {
    if (reg.hasValue()) {
        return extractTag(reg.valueReg(), scratch);
    }
    mov(ImmWord(MIRTypeToTag(reg.type())), scratch);
    return scratch;
}
```

The problem is the last part: the ARM64 backend assumes ValueTags are sign-extended, as explained [by this long comment](#). The code above doesn't sign-extend the tag. The result is that we load an object tag into the register, then we do `branchTestNumber` under `guardTypeSet`, and due to the tag not being sign-extended, we incorrectly let an object pass through the type barrier...

How to fix this: `extractTag` on a known-type seems kind of silly and suboptimal to begin with. I have a patch to specialize `guardTypeSet` for `TypedOrValueRegister`, I think we should land that. Then in a later patch we can remove this `extractTag` overload - this way it's really hard to figure out what the problem is just from the patch.

status-firefox76: --- → affected


status-firefox77: --- → affected

status-firefox-esr68: --- → affected

tracking-firefox77: --- → ?

tracking-firefox78: --- → ?

Keywords: sec-high


**Jan de Mooij** [jandem]

Comment 6 • 3 years ago

Assignee

—











Attached file [Bug 1640737 part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. ?!ain!](#) — Details

**Jan de Mooij** [jandem]

Comment 7 • 3 years ago

Assignee

—

Attached file Bug 1640737 part 2 - Add a test, remove now unused code. r?iain! — Details		
Depends on D77008		
	Pascal Chevrel:pascalc Comment 8 • 3 years ago	<div>—</div>
Jan, are you asking for tracking for 77 because you want to uplift the fix to mozilla-release and esr68 before we ship? (we ship 77 next Tuesday)		
Flags: needinfo?(jdemooij)		
	Jan de Mooij [jandem] Assignee Comment 9 • 3 years ago	<div>—</div>
(In reply to Pascal Chevrel:pascalc from comment #6)		
Jan, are you asking for tracking for 77 because you want to uplift the fix to mozilla-release and esr68 before we ship? (we ship 77 next Tuesday)		
I just requested tracking to get this bug on your radar. It would be nice to have this fixed for 77 though.		
Flags: needinfo?(jdemooij)		
	Pascal Chevrel:pascalc Comment 10 • 3 years ago	<div>—</div>
I think this can wait the next 78 cycle as we are building 77RC2 today.		
status-firefox76: affected → wontfix status-firefox77: affected → wontfix tracking-firefox77: ? → -		
	mlfbrown Comment 11 • 3 years ago	<div>—</div>
Super interesting. This test case was generated by Fuzzilli, by the way!		
	Ryan VanderMeulen [:RyanVM] Updated • 3 years ago	<div>—</div>
tracking-firefox78: ? → + tracking-firefox-esr68: --- → 78+		
	Jan de Mooij [jandem] Assignee Comment 12 • 3 years ago	<div>—</div>
Comment on attachment 9152016 [details] Bug 1640737 part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iain!		
Security Approval Request <ul style="list-style-type: none"> • How easily could an exploit be constructed based on the patch?: Difficult. (It's not clear from the patch that it's ARM64-specific, this patch doesn't touch the buggy code.) • Do comments in the patch, the check-in comment, or tests included in the patch paint a bulls-eye on the security problem?: No • Which older supported branches are affected by this flaw?: All • If not all supported branches, which bug introduced the flaw?: None • Do you have backports for the affected branches?: No • If not, how different, hard to create, and risky will they be?: Should apply or be easy to backport. • How likely is this patch to cause regressions; how much testing does it need?: Not very likely. A green Try run should be sufficient. 		
Attachment #9152016 - Flags: sec-approval?		
	Steven DeTar [:sdetar] Updated • 3 years ago	<div>—</div>
Severity: -- → S2 Priority: -- → P1		
	Daniel Veditz [:dveditz] Comment 13 • 3 years ago	<div>—</div>
Comment on attachment 9152016 [details] Bug 1640737 part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iain!		
sec-approval=dveditz		
Attachment #9152016 - Flags: sec-approval? → sec-approval+		
	Jan de Mooij [jandem] Assignee Comment 14 • 3 years ago	<div>—</div>
https://hg.mozilla.org/integration/autoland/rev/e816389af5d2c068cbb337b4e87efe568260acd4		
I'll request ESR68 approval soon.		
Flags: needinfo?(jdemooij)		
	Sebastian Hengst [:aryx] (needinfo me if it's about an intermittent or bailout) Comment 15 • 3 years ago	<div>—</div>
https://hg.mozilla.org/mozilla-central/rev/e816389af5d2		
Group: javascript-core-security → core-security-release Status: ASSIGNED → RESOLVED		

Jan de Mooij [:jandem]

Assignee

Comment 16 • 3 years ago

Comment on [attachment 9152016](#) [details]

~~Bug 1640737~~ part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iain!

ESR Uplift Approval Request

- **If this is not a sec{high,crit} bug, please state case for ESR consideration:**
- **User impact if declined:** Crashes or security bugs on ARM64 platforms.
- **Fix Landed on Version:** 78
- **Risk to taking this patch:** Low
- **Why is the change risky/not risky? (and alternatives if risky):** Already landed on Nightly. Patch applies to ESR 68.
- **String or UUID changes made by this patch:** None

Flags: ~~needinfo?(jdemooij)~~

[Attachment #9152016](#) - Flags: approval-mozilla-esr68?

Brindusa Tot[:brindusat]

Updated • 3 years ago

Flags: qe-verify-

Whiteboard: [post-critsmash-triage]

Release mgmt bot [:suhaib / :marco / :calixte]

Comment 17 • 3 years ago

As part of a security bug pattern analysis, we are requesting your help with a high level analysis of this bug. It is our hope to develop static analysis (or potentially runtime/dynamic analysis) in the future to identify classes of bugs.

Please visit [this google form](#) to reply.

Flags: needinfo?(jdemooij)

Whiteboard: [post-critsmash-triage] → [post-critsmash-triage][sec-survey]

Deian Stefan

Reporter

Comment 18 • 3 years ago

Hey folks, can we tag this bug for a bounty? If awarded we'd like to donate the cash to, say NAACP Legal Defense and Educational Fund or ACLU. Thanks!

Jan de Mooij [:jandem]

Assignee

Comment 19 • 3 years ago

I'll set the sec-bounty flag. It can take some time for the security team to make the decision on that.

Flags: sec-bounty?

Jan de Mooij [:jandem]

Assignee

Updated • 3 years ago

Flags: ~~needinfo?(jdemooij)~~

Ryan VanderMeulen [:RyanVM]

Comment 20 • 3 years ago

Comment on [attachment 9152016](#) [details]

~~Bug 1640737~~ part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iain!

Approved for 68.10esr.

[Attachment #9152016](#) - Flags: approval-mozilla-esr68? → approval-mozilla-esr68+

Ryan VanderMeulen [:RyanVM]

Comment 21 • 3 years ago

uplift

<https://hg.mozilla.org/releases/mozilla-esr68/rev/410c3a698d84a90f70c831210f6c5fa52d544b55>

[status-firefox-esr68: affected](#) → [fixed](#)

Andreea Pavel [:apavel]

Comment 22 • 3 years ago

Backed out for build bustages at MacroAssembler.cpp

Push with failures: <https://treeherder.mozilla.org/#/jobs?repo=mozilla-esr68&resultStatus=testfailed%2Cbusted%2Cexception&selectedTaskRun=AR0piX-oQk2CfLJYG-eBQ-0>

Failure log: https://treeherder.mozilla.org/logviewer.html#/jobs?job_id=305056264&repo=mozilla-esr68&lineNumber=2559

Backout: <https://hg.mozilla.org/releases/mozilla-esr68/rev/2790572dd9f56fcb8cd841fb41f93cec041e7d61>

[status-firefox78: fixed](#) → [affected](#)

Flags: needinfo?(jdemooij)

Jan de Mooij [:jandem]

Assignee

Comment 23 • 3 years ago

Attached patch [Patch for ESR68](#) — [Details](#) — [Splinter Review](#)



We just had to include the `TypeSet::PrimitiveType` function that was added after ESR68. I confirmed this builds and passes jit-tests on ESR68.

Flags: ~~needinfo?(jdemooij)~~



Jan de Mooij [:jandem] Assignee

Updated • 3 years ago



Flags: needinfo?(ryanvm)



Julien Cristau [:jcristau]

Comment 24 • 3 years ago



Looks like [comment 24](#) flipped the wrong flag.

[status-firefox78: affected](#) → [fixed](#)

[status-firefox-esr68: fixed](#) → [affected](#)



Ryan VanderMeulen [:RyanVM]

Updated • 3 years ago



Flags: ~~needinfo?(ryanvm)~~

[Attachment #9154482](#) - Flags: [approval-mozilla-esr68+](#)



Ryan VanderMeulen [:RyanVM]

Updated • 3 years ago



[Attachment #9152016](#) - Flags: ~~[approval-mozilla-esr68+](#)~~



Ryan VanderMeulen [:RyanVM]

Comment 25 • 3 years ago

[uplift](#)



<https://hg.mozilla.org/releases/mozilla-esr68/rev/47e62b30b420424780a69b7fc9e2fa07c486e8b1>

[status-firefox-esr68: affected](#) → [fixed](#)



Daniel Veditz [:dveditz]

Updated • 3 years ago



Flags: [sec-bounty?](#) → [sec-bounty+](#)



Ryan VanderMeulen [:RyanVM]

Updated • 3 years ago



[status-firefox-esr78: ---](#) → [fixed](#)

[tracking-firefox-esr78: ---](#) → [78+](#)



Tom Ritter [:tjr]

Updated • 3 years ago



Whiteboard: [\[post-critsmash-triage\]\[sec-survey\]](#) → [\[post-critsmash-triage\]\[sec-survey\]\[adv-main78+\]](#)



Tom Ritter [:tjr]

Updated • 3 years ago



Whiteboard: [\[post-critsmash-triage\]\[sec-survey\]\[adv-main78+\]](#) → [\[post-critsmash-triage\]\[sec-survey\]\[adv-main78+\]\[adv-esr68.10+\]](#)



Tom Ritter [:tjr]

Comment 26 • 3 years ago



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






Tom Ritter [:tjr]

Updated • 3 years ago



Alias: [CVE-2020-12417](#)

 Sebastian Hengst [aryx] (needinfo me if it's about an intermittent or backout) Comment 27 • 2 years ago	<div>—</div>
part 2 - Add a test, remove now unused code. r=iain https://hg.mozilla.org/integration/autoland/rev/d818a9726c698c464396d16f43fc73027a78ee4e https://hg.mozilla.org/mozilla-central/rev/d818a9726c69	
 Julien Cristau [jcristau] Updated • 2 years ago	<div>—</div>
Flags: in-testsuite+	
 Daniel Veditz [dveditz] Updated • 2 years ago	<div>—</div>
Group: core-security-release	

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

Top ↑