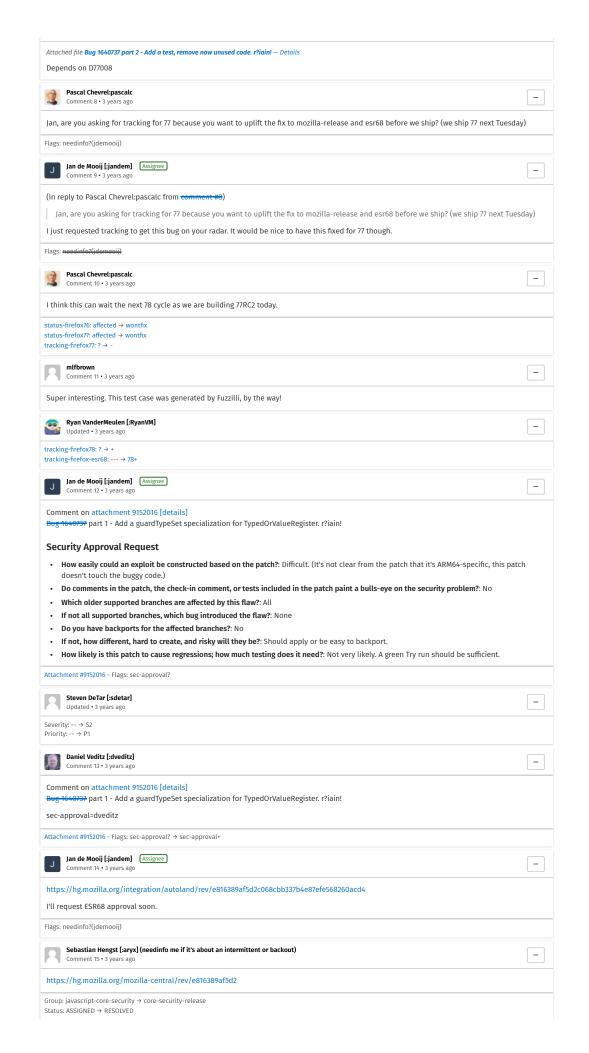
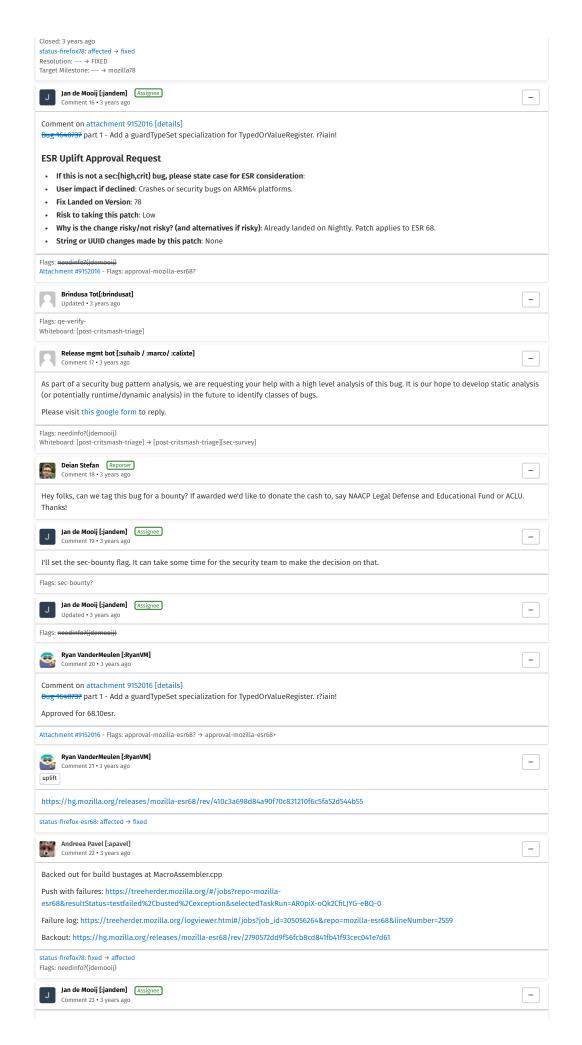
```
main();
  gc();
Daniel Veditz [:dveditz]
Updated • 3 years ago
                                                                                                                                          -
Group: core-security → javascript-core-security
      Jan de Mooij [:jandem] Assignee
                                                                                                                                           _
The two tests are identical AFAICS?
      Deian Stefan Reporter
                                                                                                                                           -
      Comment 2 • 3 years ago
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] Assignee
                                                                                                                                           _
      Comment 3 • 3 years ago
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] Assignee
                                                                                                                                          -
I think I know what's going on here. Really good find.
Assignee: nobody → jdemooij
Status: NEW → ASSIGNED
      Jan de Mooij [:jandem] 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); \\
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] Assignee
                                                                                                                                           -
Attached file Bug 1640737 part 1 - Add a guardTypeSet specialization for TypedOrValueRegister. r?iain! — Details
      Jan de Mooij [:jandem] Assignee
```

Comment 7 • 3 years ago





Attache	ed patch Patch for ESR68 — Details — Splinter Review	
We jus	it had to include the TypeSet::PrimitiveType function that was added after ESR68. I confirmed this builds and passes jit-tests o .	n
Flags: e	ecdinfo:/(jdemeoij)	
	lan de Mooij [.jandem]	
	Updated • 3 years ago eedinfo?(ryanvm)	
	ulien Cristau	
	Comment 24 • 3 years ago	
Looks	like comment 22 flipped the wrong flag.	
	irefox78: affected → fixed irefox-esr68: fixed → affected	
	Ryan VanderMeulen [:RyanVM]	
	Updated • 3 years ago	
	ent #9154482 - Flags: approval-mozilla-esr68+	
	Ryan VanderMeulen [:RyanVM] Updated • 3 years ago	
Attachn	nent #9152016 - Flags: approval-mozilla-occ68	
	Ryan VanderMeulen [:RyanVM]	
uplift	Comment 25 • 3 years ago	
https:	//hg.mozilla.org/releases/mozilla-esr68/rev/47e62b30b420424780a69b7fc9e2fa07c486e8b1	
status-f	irefox-esr68: affected → fixed	
	Daniel Veditz [:dveditz] Updated • 3 years ago	
	ec-bounty? → sec-bounty+	
60	Ryan VanderMeulen [:RyanVM]	
	Updated • 3 years ago	
	-firefox-es/78: → 78+	
	Tom Ritter [tjr] Updated • 3 years ago	
	oard: [post-critsmash-triage][sec-survey] → [post-critsmash-triage][sec-survey][adv-main78*]	
	Tom Ritter [:tir]	
	Updated • 3 years ago vard: [post-critsmash-triage][sec-survey][adv-main78+] → [post-critsmash-triage][sec-survey][adv-main78+][adv-esr68.10+]	
E.B.	Tom Ritter [:tjr]	
	Comment 26 • 3 years ago	
Attache	d file advisory.txt — Details	
	Tom Dittor (stir)	
	Tom Ritter [:tjr] Updated • 3 years ago	

Sebastian Hengst [:aryx] (needinfo me if it's about an intermittent or backout) Comment 27 • 2 years ago	_
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	_
Flags: in-testsuite+	
Daniel Veditz [:dveditz] Updated • 2 years ago	_
Group: core-security-release	

You need to \log in before you can comment on or make changes to this bug.

Top ↑