

ClosedBug 1709976 (CVE-2021-29970)Opened 2 years agoClosed 2 years ago

AddressSanitizer: heap-use-after-free in [@@ mozilla::PresShell::RemoveRefreshObserver]

Categories

Product: Core ▼
Component: Disability Access APIs ▼

Type: defect
Priority: Not setSeverity: --

Tracking

Status: VERIFIED FIXED
Milestone: 91 Branch

Tracking Flags:
firefox-esr78
firefox88
firefox89
firefox90
firefox91

Tracking
90+

+
+

Status
verified
wontfix
wontfix
verified
verified

People

(Reporter: sourc7, Assigned: eeejay)

Details

(Keywords: csetype-uaf, sec-high, Whiteboard: [reporter-external] [client-bounty-form] [verif?][sec-survey][adv-main90+][adv-esr78.12+])

Crash Data

Attachments

asan.archlinux.txt
2 years ago Irvan Kurniawan (sourc7)
42.35 KB, text/plain

Details

asan.windows10.txt
2 years ago Irvan Kurniawan (sourc7)
13.30 KB, text/plain

Details

Firefox - UAF in DocAccessible::Shutdown - e10s disabled.mp4
2 years ago Irvan Kurniawan (sourc7)
1.63 MB, video/mp4

Details

asan.e10senabled.txt
2 years ago Irvan Kurniawan (sourc7)
20.48 KB, text/plain

Details

asan.e10sdisabled.txt
2 years ago Irvan Kurniawan (sourc7)
15.23 KB, text/plain

Details

valgrind.txt
2 years ago Irvan Kurniawan (sourc7)
19.37 KB, text/plain

Details

Bug 1709976 - Remove selection listeners from shutting down PresShell. r?jamie
2 years ago Eitan Isaacson [eeejay]
48 bytes, text/x-phabricator-request

tjr : approval-mozilla-beta+
tjr : approval-mozilla-esr78+
tjr : sec-approval+

Details | Review

launcher.bundle.html
2 years ago Irvan Kurniawan (sourc7)
643 bytes, text/html

Details

testcase.main.html
2 years ago Irvan Kurniawan (sourc7)
247 bytes, text/html

Details

Firefox - UAF in DocAccessibleShutdown with one-click - Windows 10 using Microsoft Narrator.mp4
2 years ago Irvan Kurniawan (sourc7)
495.23 KB, video/mp4

Details

Firefox 89.0.2 - RemoveRefreshObserver Crash.mp4
2 years ago Irvan Kurniawan (sourc7)
785.93 KB, video/mp4

Details

advisory.txt
2 years ago Tom Ritter [tjr]
258 bytes, text/plain

Details

Show Obsolete

Bottom ▼Tags ▼Timeline ▼

Irvan Kurniawan (sourc7) Reporter
Description • 2 years ago

Attached file asan.archlinux.txt — Details

While fuzzing on Firefox ASan fuzzing build, the tab crashed on Arch Linux with SUMMARY: AddressSanitizer: heap-use-after-free RefPtr.h:286:27 in get and Windows 10 with heap-use-after-free gecko/layout/base/PresShell.cpp:9914 in mozilla::PresShell::RemoveRefreshObserver
The testcase is still intermittent it require FuzzingFunctions.enableAccessibility() function and few interaction. Currently I'm still refactor the code in hope it able to reproduce easily. I'll attach the testcase once I'm done with it =).

Reproduced on

- Firefox Nightly 90.0a1 (m-c-20210506092558-fuzzing-asan-opt) (64-bit) on Arch Linux

- Firefox Nightly 90.0a1 (m-c-20210506214311-fuzzing-asan-opt) (64-bit) on Windows 10

ASan

Arch Linux

=====

==1607538==ERROR: AddressSanitizer: heap-use-after-free on address 0x62200051170 at pc 0x7fbf691d93a9
READ of size 8 at 0x62200051170 thread T0 (Web Content)

#0 0x7fbf691d93a8 in get /builds/worker/workspace/obj-build/dist/include/mozilla/RefPtr.h:286:27

#1 0x7fbf691d93a8 in operator nsPresContext * /builds/worker/workspace/obj-build/dist/include/mozilla/PresSh

#2 0x7fbf691d93a8 in GetPresContext /builds/worker/workspace/obj-build/dist/include/mozilla/PresSh

#3 0x7fbf691d93a8 in mozilla::PresShell::RemoveRefreshObserver(nsARefreshObserver*, mozilla::Flush

#4 0x7fbf6c554c63 in mozilla::a11y::NotificationController::Shutdown() /builds/worker/checkouts/ge

#5 0x7fbf6c5a1ef in mozilla::a11y::DocAccessible::Shutdown() /builds/worker/checkouts/gecko/acces

#6 0x7fbf6c5ab4ed in Unlink /builds/worker/checkouts/gecko/accessible/generic/LocalAccessible.cpp:

#7 0x7fbf6c5ab4ed in mozilla::a11y::DocAccessible::cycleCollection::Unlink(void*) /builds/worker/c

#8 0x7fbf6128d1e5 in nsCycleCollector::CollectWhite() /builds/worker/checkouts/gecko/xpcom/base/ns

#9 0x7fbf6128ffa3 in nsCycleCollector::Collect(ccType, js::SliceBudget&, nsICycleCollectorListener

#10 0x7fbf6129319d in nsCycleCollector_collect(nsICycleCollectorListener*) /builds/worker/checkout

#11 0x7fbf64a42e5e in nsJSContext::CycleCollectNow(nsICycleCollectorListener*) /builds/worker/chec

#12 0x7fbf66081c82 in mozilla::dom::FuzzingFunctions_Binding::cycleCollect(JSContext*, unsigned in

#13 0x7fbf6d0e0622 in CallJSNative /builds/worker/checkouts/gecko/js/src/vm/Interpreter.cpp:427:13

#14 0x7fbf6d0e0622 in js::InternalCallOrConstruct(JSContext*, JS::CallArgs const&, js::MaybeConstr

#15 0x7fbf6d0cd7c5 in CallFromStack /builds/worker/checkouts/gecko/js/src/vm/Interpreter.cpp:576:1

#16 0x7fbf6d0cd7c5 in Interpret(JSContext*, js::RunState&) /builds/worker/checkouts/gecko/js/src/v

#17 0x7fbf6d0b6e36 in js::RunScript(JSContext*, js::RunState&) /builds/worker/checkouts/gecko/js/s

#18 0x7fbf6d0e615b in js::InternalCallOrConstruct(JSContext*, JS::CallArgs const&, js::MaybeConstr

#19 0x7fbf6d0e7d5b in js::Call(JSContext*, JS::Handle<JS::Value>, JS::Handle<JS::Value>, js::AnyIn

#20 0x7fbf6d953f62 in JS::Call(JSContext*, JS::Handle<JS::Value>, JS::Handle<JS::Value>, JS::Handl

◀

▶

Windows 10

=====

==3420==ERROR: AddressSanitizer: heap-use-after-free on address 0x129b11675970 at pc 0x7ffd203f44df bp
READ of size 8 at 0x129b11675970 thread T0

#0 0x7ffd203f44de in mozilla::PresShell::RemoveRefreshObserver /builds/worker/checkouts/gecko/layo

#1 0x7ffd2359999a in mozilla::a11y::NotificationController::Shutdown /builds/worker/checkouts/geck

#2 0x7ffd23619b22 in mozilla::a11y::DocAccessible::Shutdown /builds/worker/checkouts/gecko/acces

#3 0x7ffd23615021 in mozilla::a11y::DocAccessible::cycleCollection::Unlink /builds/worker/checkout

#4 0x7ffd165b5580 in nsCycleCollector::CollectWhite /builds/worker/checkouts/gecko/xpcom/base/nsCy

#5 0x7ffd165b8abf in nsCycleCollector::Collect /builds/worker/checkouts/gecko/xpcom/base/nsCycleCo

#6 0x7ffd165bce7b in nsCycleCollector_collect /builds/worker/checkouts/gecko/xpcom/base/nsCycleCol

#7 0x7ffd1a4c82ef in nsJSContext::CycleCollectNow /builds/worker/checkouts/gecko/dom/base/nsJSEnvi

#8 0x7ffd1c435b20 in mozilla::dom::FuzzingFunctions_Binding::cycleCollect /builds/worker/workspac

#9 0x7ffd24373705 in js::InternalCallOrConstruct /builds/worker/checkouts/gecko/js/src/vm/Interpre

#10 0x7ffd2435fbbe in Interpret /builds/worker/checkouts/gecko/js/src/vm/Interpreter.cpp:3227

#11 0x7ffd2434569a in js::RunScript /builds/worker/checkouts/gecko/js/src/vm/Interpreter.cpp:396

#12 0x7ffd243739f5 in js::InternalCallOrConstruct /builds/worker/checkouts/gecko/js/src/vm/Interpr

#13 0x7ffd24376368 in js::Call /builds/worker/checkouts/gecko/js/src/vm/Interpreter.cpp:589

#14 0x7ffd24db97fb in JS::Call /builds/worker/checkouts/gecko/js/src/jsapi.cpp:2849

#15 0x7ffd1c1194cd in mozilla::dom::EventHandlerNonNull::Call /builds/worker/workspace/obj-build/d

#16 0x7ffd1d3521c6 in mozilla::JSEventHandler::HandleEvent /builds/worker/checkouts/gecko/dom/even

#17 0x7ffd1d3085e0 in mozilla::EventListenerManager::HandleEventSubType /builds/worker/checkouts/g

#18 0x7ffd1d30a34c in mozilla::EventListenerManager::HandleEventInternal /builds/worker/checkouts/g


#19 0x7ffd1d2ef856 in mozilla::EventTargetChainItem::HandleEvent /builds/worker/checkouts/gecko/do

#20 0x7ffd1d2ed9da in mozilla::EventTargetChainItem::HandleEventTargetChain /builds/worker/checkou

◀

▶

Flags: sec-bounty?




Irvan Kurniawan (sourc7)

Reporter

Comment 1 • 2 years ago

Attached file [asan.windows10.txt](#) — Details



Andrew McCreight [:mccr8]

Comment 2 • 2 years ago

It looks like NotificationController::Shutdown is being called on a freed mPressShell.

Group: firefox-core-security → dom-core-security



Andrew McCreight [:mccr8]
Comment 3 • 2 years ago • [Edited](#)



Just looking at the stack, it kind of looks like mPresShell should be rooted on this line in nsDocumentViewer::LoadComplete:
`EventDispatcher::Dispatch(window, mPresContext, &event, nullptr, &status);` . There's an earlier scope in this method that holds a strong stack ref to the pres shell with the comment "Hold strong ref because this could conceivably run script" but I'm not sure why it wouldn't be needed here.



James Teh [:jamie]
Comment 4 • 2 years ago



What puzzles me about this is that PresShell::Destroy should call DocAccessible::Shutdown. So, if NotificationController has a freed PresShell, that means PresShell::Destroy never got called? Or a DocAccessible was somehow created with a freed PresShell? Or a DocAccessible was created with a PresShell after the PresShell was destroyed (but before it was freed)?

It's probably worth at least adding an assertion to the DocAccessible constructor to ensure it never gets passed a destroyed PresShell.

See also [bug 1502338](#).



Andrew McCreight [:mccr8]
Comment 5 • 2 years ago



It is possible that the test case is doing something weird like calling `FuzzingFunctions.enableAccessibility()` during an event or something. Hopefully if we get a test case it will shed some light on the situation.



Irvan Kurniawan (:sourc7) Reporter
Comment 6 • 2 years ago



Alright I've refactored the code, it require responsive design mode + multiple fast click to trigger the UAF. I think it's still possible to trigger the UAF without RDM which is interesting to find out.

There are 3 ways to trigger the crash, with `./mach run` (e10s enabled) and `./mach run --disable-e10s` (e10s disabled). On e10s disabled, it doesn't require launcher or reload to trigger the crash (which is more easy to trigger).

Steps to Reproduce:

e10s enabled (with launcher):

1. Open Firefox with fuzzing build (`ac_add_options --enable-fuzzing`)
2. Go to `about:config`
3. Set `fuzzing.enabled` to `true`
4. Set `dom.disable_open_during_load` to `false`
5. Restart Firefox
6. Visit attached `launcher.bundle.html`
7. Click "Launch" button
8. Repeatedly click the text then simultaneously press `ctrl+shift+m` every `-2s`
9. After repeated tries, the tab is crashed

e10s enabled (close the tab):

1. Open Firefox with fuzzing build (`ac_add_options --enable-fuzzing`)
2. Go to `about:config`
3. Set `fuzzing.enabled` to `true`
4. Restart Firefox
5. Visit attached `testcase.main.html`
6. Click "Launch" button
7. Repeatedly click the text then simultaneously press `ctrl+shift+m` every `-2s`
8. After trying for a few seconds, close the tab
9. ASan will show `heap-use-after-free`

e10s disabled:

1. Open Firefox with fuzzing build (`ac_add_options --enable-fuzzing`) then `./mach run --disable-e10s`
2. Go to `about:config`
3. Set `fuzzing.enabled` to `true`
4. Visit attached `testcase.main.html`
5. Repeatedly click the text then simultaneously press `ctrl+shift+m` every `-2s`
6. The tab is crashed.

(I also will attach steps to reproduce video to demonstrate the crash)



Irvan Kurniawan (:sourc7) Reporter
Comment 7 • 2 years ago



Attached file [launcher.bundle.html](#) (obsolete) — [Details](#)



Irvan Kurniawan (:sourc7) Reporter
Comment 8 • 2 years ago



Attached file [testcase.reload.html](#) (obsolete) — [Details](#)



Irvan Kurniawan (:sourc7) Reporter
Comment 9 • 2 years ago



Attached file [testcase-main.html](#) (obsolete) — Details



Irvan Kurniawan (:sourc7)
Comment 10 • 2 years ago

Reporter



Attached video [Firefox - UAF in DocAccessible::Shutdown - e10s enabled launcher.mp4](#) (obsolete) — Details

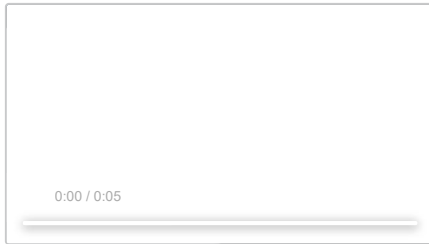


Irvan Kurniawan (:sourc7)
Comment 11 • 2 years ago

Reporter



Attached video [Firefox - UAF in DocAccessible::Shutdown - e10s disabled.mp4](#) — Details

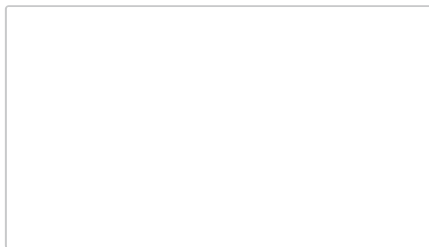


Irvan Kurniawan (:sourc7)
Comment 12 • 2 years ago

Reporter



Attached file [asan.e10senabled.txt](#) — Details

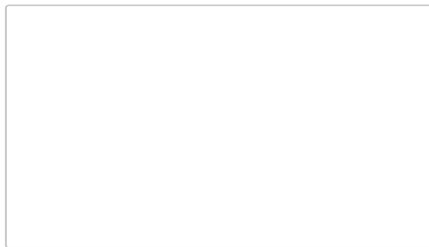


Irvan Kurniawan (:sourc7)
Comment 13 • 2 years ago

Reporter



Attached file [asan.e10sdisabled.txt](#) — Details



Daniel Veditz [:dveditz]
Comment 14 • 2 years ago



This could be more of a fuzzing harness problem than something that could be reproduced in the wild, given that fuzzing hook function wasn't really designed to be called at random times repeatedly, and wouldn't be in a real web page.

but needs more investigation to be sure of that.



Irvan Kurniawan (:sourc7)
Comment 15 • 2 years ago

Reporter



(In reply to Daniel Veditz [:dveditz] from [comment #14](#))

This could be more of a fuzzing harness problem than something that could be reproduced in the wild, given that fuzzing hook function wasn't really designed to be called at random times repeatedly, and wouldn't be in a real web page.

but needs more investigation to be sure of that.

I've installed NVDA on Windows 10, then run the launcher.bundle.html (with steps to reproduce above) on Firefox 90.0a1 (2021-05-12) (64-bit) official build, the Firefox crashes on the first try with signature mozilla::PresShell::RemoveRefreshObserver (as on attached ASan stack above).

Hereby the crash reports: <https://crash-stats.mozilla.org/report/index/4bad5f58-da2c-4262-b017-7f82a0210512>



Irvan Kurniawan (:sourc7)
Comment 16 • 2 years ago

Reporter



In the crash report above, one of the CPU registers shows "0xe5e5e5e5e5e5e5" indicating UAF.



Irvan Kurniawan (:sourc7)
Comment 17 • 2 years ago

Reporter



I can also reproduce this on Arch Linux with [Orca screen reader](#) activated using Firefox Nightly 90.0a1 (2021-05-17) (64-bit) (Official Build) then run launcher.bundle.html without FuzzingFunctions.enableAccessibility() on the code.

Interestingly I got 3 different crash reports as follows:

1. [[@ nsRefreshDriver::RemoveRefreshObserver](#)] SIGSEGV at 0xe5e5e5e5e5e5e5e5 -> 0x0
2. [[@ nsRefreshDriver::RemoveRefreshObserver](#)] SIGSEGV /SEGV_MAPERR at 0xcad0000000
3. [[@ nsFrameSelection::GetSelection](#)] SIGSEGV /SEGV_MAPERR at 0x40db1bf0



Irvan Kurniawan (sourc7)

Reporter

Comment 18 • 2 years ago



(In reply to Irvan Kurniawan (sourc7) from [comment #15](#))

I've installed NVDA on Windows 10, then run the `launcher.bundle.html` (with steps to reproduce above) on Firefox 90.0a1 (2021-05-12) (64-bit) official build, the Firefox crashes on the first try with signature `mozilla::PresShell::RemoveRefreshObserver` (as on attached ASan stack above).

It turns out that I also able to reproduce this when using [Microsoft Narrator](#) (Windows built in screen reader).

I'm sure that as long the Accessibility shows `Activated: true` on `about:support` it able to reproduce on Firefox official build (same as using `FuzzingFunctions.enableAccessibility()` on Firefox fuzzing build)



James Teh (Jamie)

Comment 19 • 2 years ago



I can't reproduce this here with the NVDA screen reader despite many attempts. :(

I can think of some assertions we could add, but that's probably not helpful because catching local build assertions in content processes is currently really difficult, at least on Windows. I guess we could land some diagnostic assertions to try to catch the problem on crash-stats, but that might expose the bug.

In addition to my thoughts in [comment 4](#), I wonder whether somehow a second `DocAccessible` is being created for the same `PresShell`. If that happens, `PresShell::SetDocAccessible` would get called, overriding its `DocAccessible`. When the `PresShell` gets destroyed, it would shut down the second `DocAccessible`, but not the first, leaving the first with a destroyed `DocAccessible`. That said, I don't see how a second `DocAccessible` could be created; we always ask the `PresShell` for its `DocAccessible` before creating a new one.

:dveditz, do you think it'd make sense to land these as `DIAGNOSTIC_ASSERTS`? If so, how should I go about doing that without exposing a potential UAF?

Flags: needinfo?(dveditz)



Irvan Kurniawan (sourc7)

Reporter

Comment 20 • 2 years ago



(In reply to James Teh (Jamie) from [comment #19](#))

In addition to my thoughts in [comment 4](#), I wonder whether somehow a second `DocAccessible` is being created for the same `PresShell`. If that happens, `PresShell::SetDocAccessible` would get called, overriding its `DocAccessible`. When the `PresShell` gets destroyed, it would shut down the second `DocAccessible`, but not the first, leaving the first with a destroyed `DocAccessible`. That said, I don't see how a second `DocAccessible` could be created; we always ask the `PresShell` for its `DocAccessible` before creating a new one.

In another testcase it throw assertion as follow `Assertion failure: !mDocument->GetPresShell() (Where did this shell come from?)`, at `layout/base/PresShell.cpp:4121`.

From the assertion message, I think it's related to this bug. I also found the assertion and UAF occur because of combination of `ctrl+shift+m` and mouse click. Without mouse click the `RemoveRefreshObserver` won't use the freed `mPressShell`.



Irvan Kurniawan (sourc7)

Reporter

Comment 21 • 2 years ago



(In reply to Irvan Kurniawan (sourc7) from [comment #20](#))

In another testcase it throw assertion as follow `Assertion failure: !mDocument->GetPresShell() (Where did this shell come from?)`, at `layout/base/PresShell.cpp:4121`.

From the assertion message, I think it's related to this bug. I also found the assertion and UAF occur because of combination of `ctrl+shift+m` and mouse click. Without mouse click the `RemoveRefreshObserver` won't use the freed `mPressShell`.

Hmm, I'm not sure whether it is related or not, hereby the signature report for [mozilla::PresShell::DoFlushPendingNotifications](#).

I think someone who knows about `PresShell` could look into this bug, hopefully it will solve this problem too.



James Teh (Jamie)

Comment 22 • 2 years ago



If you're able to see assertions, perhaps I can provide a patch which adds assertions I'd like to verify for the accessibility crash?

```
diff --git a/accessible/generic/DocAccessible.cpp b/accessible/generic/DocAccessible.cpp
index c7411cb66f724..e19674c6d9556 100644
--- a/accessible/generic/DocAccessible.cpp
+++ b/accessible/generic/DocAccessible.cpp
@@ -98,6 +98,10 @@ DocAccessible::DocAccessible(dom::Document* aDocument,
    mDoc = this;

    MOZ_ASSERT(mPresShell, "should have been given a pres shell");
+   MOZ_ASSERT(!mPresShell->IsDestroying(),
+              "Should never get a destroying PresShell");
+   MOZ_ASSERT(!mPresShell->GetDocAccessible(),
+              "PresShell shouldn't already have a DocAccessible");
    mPresShell->SetDocAccessible(this);
}
```



Irvan Kurniawan (sourc7)

Reporter

Comment 23 • 2 years ago



(In reply to James Teh (Jamie) from [comment #22](#))

If you're able to see assertions, perhaps I can provide a patch which adds assertions I'd like to verify for the accessibility crash?

```
diff --git a/accessible/generic/DocAccessible.cpp b/accessible/generic/DocAccessible.cpp
index c7411cb66f724..e19674c6d9556 100644
--- a/accessible/generic/DocAccessible.cpp
+++ b/accessible/generic/DocAccessible.cpp
@@ -98,6 +98,10 @@ DocAccessible::DocAccessible(dom::Document* aDocument,
    mDoc = this;

    MOZ_ASSERT(mPresShell, "should have been given a pres shell");
+   MOZ_ASSERT(!mPresShell->IsDestroying(),
+               "Should never get a destroying PresShell");
+   MOZ_ASSERT(!mPresShell->GetDocAccessible(),
+               "PresShell shouldn't already have a DocAccessible");
    mPresShell->SetDocAccessible(this);
}
```

Sorry I forgot to add, the assertion on [comment 20](#) is from another testcase, it not required Accessibility and more easily triggered just by switch to Responsive Design Mode (ctrl+shift+m) then click the text. I'm still not sure whether that related or not, but it give a clue as `PresShell` is also on the stack.

Thanks for the patch, I've added both assertion to my Firefox ASan build, but unfortunately the browser still crashes with heap-use-after-free.



James Teh [Jamie]

Comment 24 • 2 years ago



Okay. Thanks for trying it. In that case, no point in landing these assertions for this bug.

Back to the drawing board.

Flags: [needinfo?\(dveditz\)](#)



James Teh [Jamie]

Comment 25 • 2 years ago



Oh, I see the assertions in [comment 20](#) are `DIAGNOSTIC_ASSERTs`. Do ASan builds have debug asserts enabled? It might be worth changing the `MOZ_ASSERTs` in my patch to `MOZ_DIAGNOSTIC_ASSERT`.



Andrew McCreight [mccr8]

Comment 26 • 2 years ago



The usual ASan builds are opt, so they don't have `MOZ_ASSERT` enabled.



Irvan Kurniawan [sourc7]

Reporter

Comment 27 • 2 years ago



(In reply to James Teh [Jamie] from [comment #25](#))

Oh, I see the assertions in [comment 20](#) are `DIAGNOSTIC_ASSERTs`. Do ASan builds have debug asserts enabled? It might be worth changing the `MOZ_ASSERTs` in my patch to `MOZ_DIAGNOSTIC_ASSERT`.

Still same, it still crash with heap-use-after-free.



Irvan Kurniawan [sourc7]

Reporter

Comment 28 • 2 years ago



Alright I've captured the UAF with RR, here the pernosco debugging session link: https://pernos.co/debug/2EryKl_CiVGymWBqd-qw-A/index.html



James Teh [Jamie]

Comment 29 • 2 years ago



Thanks.

Eitan, Pernosco is still mostly unusable with a screen reader. :(Would you mind taking a look at this?

Flags: [needinfo?\(eitan\)](#)



Eitan Isaacson [eeejay]

Assignee

Comment 30 • 2 years ago



A `DocAccessible` is not supposed to outlive its `PresShell`. The `DocAccessible` indirectly holds a weak ref to the `PresShell` (via `NotificationController`). When the doc is constructed, it passes its pointer to its `PresShell` so that when the `PresShell` is destroyed it will shutdown the doc, and thus a weak reference is assumed to be safe.

This crash happens when a doc is shutdown after its `presshell` is destroyed. This happens because a doc is constructed, and sets its reference on its `presshell` via `SetDocAccessible`, but then when the doc is inserted into its `outerdoc`, the previous doc is shutdown, and calls `SetDocAccessible` with `nullptr` on the same `PresShell`. So the `PresShell` never has a valid reference to the current `DocAccessible`. This typically is not a problem unless the `PresShell` is destroyed before the `DocAccessible` is shutdown, which is what happens here.

Flags: [needinfo?\(eitan\)](#)



James Teh [Jamie]

Comment 31 • 2 years ago



(In reply to Eitan Isaacson [eeejay] from [comment #30](#))

but then when the doc is inserted into its `outerdoc`, the previous doc is shutdown,

But how did we create a new doc for that `PresShell` without shutting the other one down first? As I understand it, that's never supposed to happen. `DocManager::GetDocAccessible` always checks for an existing `DocAccessible` on the `PresShell` first.

Even if a doc is bound to its `OuterDoc` late, we just bind the doc; we don't create a new one.

Flags: needinfo?(eitan)

Eitan Isaacson [eejay] Assignee Comment 32 • 2 years ago -

A DOM document is not guaranteed to have the same presshell thru its entire lifetime, see `Document::CreatePresShell` and `Document::DeletePresShell`. So if the DocManager tries to find a DocAccessible with a Document that had its PresShell swapped, it will fail. This will create a new DocAccessible.

Flags: ~~needinfo?(eitan)~~


Eitan Isaacson [:eeejay] Assignee
Comment 33 • 2 years ago
—

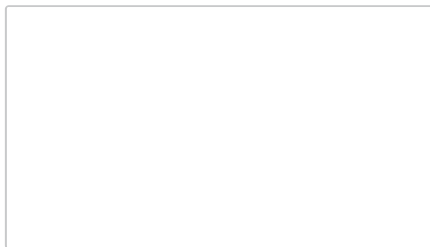
To add to that, you would think `PresShell::Destroy` would be called on the old `presshell` to shutdown the previous `DocAccessible`, but that seems to not be the case.

 **James Teh** [:jamie]
Comment 34 • 2 years ago

Right... and that's the really problematic piece here. If that's the cause of the problem, why isn't `PresShell::Destroy` being called? And if that's intentional, where else are we supposed to shut down the `DocAccessible`?


 Ivan Kurniawan (:sour?) Reporter
Comment 35 • 2 years ago

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



From Valgrind report it show 2 invalid read and 1 invalid write on following function:

- Invalid read of size 8 on mozilla::PresShell::RemoveRefreshObserver -> GetPresContext
- Invalid read of size 8 on mozilla::a11y::SelectionManager::RemoveDocSelectionListener -> nsFrameSelection
- Invalid write of size 8 on mozilla::a11y::DocAccessible::Shutdown() -> SetDocAccessible

 **James Teh** [:jamie]
Comment 36 • 2 years ago

(In reply to Eitan Isaacson [::eejay] from [comment #39](#))

This happens because a doc is constructed, and sets its reference on its presshell via `SetDocAccessible`

If there were a previous doc, this should have triggered the assertion I suggested adding in [comment 22](#), but it didn't.

We could also try this assertion, but I don't think it'll help if the other one didn't:

```
diff --git a/accessible/generic/DocAccessible.cpp b/accessible/generic/DocAccessible.cpp
index e19674c6d9556..20486bf7fdc9 100644
--- a/accessible/generic/DocAccessible.cpp
+++ b/accessible/generic/DocAccessible.cpp
@@ -415,6 +415,8 @@ void DocAccessible::Shutdown() {
     MOZ_ASSERT(!mParent, "Parent has to be null!");
 }

+ MOZ_ASSERT(mPresShell->GetDocAccessible() == this,
+ "PresShell should reference this DocAccessible");
 mPresShell->SetDocAccessible(nullptr);
 mPresShell = nullptr; // Avoid reentrancy
```

 James Teh [jamie]
Comment 37 • 2 years ago

(Of course, if testing with an opt build, that should be MOZ_DIAGNOSTIC_ASSERT)

Irvan Kurniawan (sour7) Reporter
Comment 38 • 2 years ago

(In reply to James Teh [;jamie] from [comment #36](#))

(In reply to Eitan Isaacson [::eejay] from [comment #30](#))

This happens because a doc is constructed, and sets its reference on its presshell via `SetDocAccessible`

If there were a previous doc, this should have triggered the assertion I suggested adding in [comment 22](#), but it didn't.

We could also try this assertion, but I don't think it'll help if the other one didn't:

```
diff --git a/accessible/generic/DocAccessible.cpp b/accessible/generic/DocAccessible.cpp
index e19674c6d9556..20486bf7fdcc9 100644
--- a/accessible/generic/DocAccessible.cpp
```

```
+++ b/accessible/generic/DocAccessible.cpp
@@ -415,6 +415,8 @@ void DocAccessible::Shutdown() {
     MOZ_ASSERT(!mParent, "Parent has to be null!");
 }

+ MOZ_ASSERT(mPresShell->GetDocAccessible() == this,
+ "PresShell should reference this DocAccessible");
+ mPresShell->SetDocAccessible(nullptr);
+ mPresShell = nullptr; // Avoid reentrancy
```

I've added the assertion using `MOZ_DIAGNOSTIC_ASSERT` on Asan build, but unfortunately it still crashes with UAF on `RemoveRefreshObserver` .



Ivan Kurniawan [sourc7]

Reporter

Comment 39 • 2 years ago



In the Arch Linux and Windows 10 when using debug build it hit assertion as follow:

Assertion failure: ObserverCount() == mEarlyRunners.Length() (observers, except pending selection scrolls, should have been unregistered), at /builds/worker/checkouts/gecko/layout/base/nsRefreshDriver.cpp:1153



Eitan Isaacson [eeejay]

Assignee

Comment 40 • 2 years ago



OK. I was just kidding about all that and I was way off. Looks like yes indeed presshell/docacc pairs get out of sync, that is why docaccessibles keep on getting created when there should already be a live one.

But i needed to figure out the origin of the mixup.. looks like during the Destroy method of a presshell, its DocAccessible is correctly shut down, but if there is a selection change that needs to get flushed it happens via `MaybeReleaseCapturingContent` in the PresShell's destroy method after the current DocAccessible was shut down. This triggers our selection manager to spawn a new DocAccessible in the Destroy phase of the presshell, and things get out of whack.

This shouldn't happen because the old DocAccessible's shutdown should remove the doc's/presshell's selection listeners. But this isn't the case because we add special control listeners via `SetControlSelectionListener` and don't clear those outside of focus changes. I think those should be cleared in doc shutdown as well.

This is a similar issue to [bug-1330739](#), I think the weak references might still be of use there in case of SelectionManager/DocManager shutdown.



Eitan Isaacson [eeejay]

Assignee

Comment 41 • 2 years ago



Attached file [Bug 1709976 - Remove selection listeners from shutting down PresShell. r?Jamie](#) — Details



Phabricator Automation

Updated • 2 years ago



Assignee: nobody → eitan
Status: NEW → ASSIGNED



Eitan Isaacson [eeejay]

Assignee

Comment 42 • 2 years ago



Since this crash is hard to reproduce, i would rely on the reporter to verify that this is actually remedied.



James Teh [Jamie]

Comment 43 • 2 years ago



(In reply to Eitan Isaacson [eeejay] from [comment #40](#))

| This triggers our selection manager to spawn a new DocAccessible in the Destroy phase of the presshell, and things get out of whack.

That was the purpose of the `mPresShell->IsDestroying()` assertion I suggested, but it turns out that `mIsDestroying` is set pretty late in `PresShell::Destroy()`, after the DocAccessible is shut down and `MaybeReleaseCapturingContent` is called. Ug.



Eitan Isaacson [eeejay]

Assignee

Comment 44 • 2 years ago



I didn't realize that was your suggestion. I thought of moving it earlier but don't know what that would screw with.



Eitan Isaacson [eeejay]

Assignee

Comment 45 • 2 years ago



Irvan,

Is there a chance you can try this patch and verify it remedies the issue?

Flags: needinfo?(susah.yak)



James Teh [Jamie]

Comment 46 • 2 years ago



(In reply to Eitan Isaacson [eeejay] from [comment #44](#))

| I didn't realize that was your suggestion.

It was earlier in the bug; see [comment 22](#). Anyway, this wouldn't have fixed the bug; it just would have made it easier to catch. :)

Thanks for figuring this out (and patching it).



Ivan Kurniawan [sourc7]

Reporter

Comment 47 • 2 years ago



(In reply to Eitan Isaacson [eeejay] from [comment #45](#))


Irvan,

Is there a chance you can try this patch and verify it remedies the issue?

Thanks! It finally solve the issue!

After applied the [patch](#) (on Arch Linux), I no longer able to reproduce the crash, and on debug build it no longer hit the assertion on [comment 39](#).

Flags: [needinfo?\(eusanys\)](#)



Irvan Kurniawan (:sourc7)

Reporter

Comment 48 • 2 years ago


Attached file [launcher.bundlewfx.html](#) (obsolete) — [Details](#)

(In reply to Eitan Isaacson [:eeejay] from [comment #42](#))

Since this crash is hard to reproduce, i would rely on the reporter to verify that this is actually remedied.

Eitan, can you reproduce with steps to reproduce below:

1. Open Microsoft Narrator
2. Open Firefox Nightly
3. Go to about:config
4. Set dom.disable_open_during_load to false (to disable pop-up blocker)
5. Visit attached launcher.bundlewfx.html
6. Click "Launch" button
7. Repeatedly fast click the text as possible (my click per second: 7) then simultaneously press ctrl+shift+m every ~1s (as on attached video below)
8. The tab will crashed.




Irvan Kurniawan (:sourc7)

Reporter

Comment 49 • 2 years ago

Attached video [Firefox - UAF in DocAccessibleShutdown - Windows 10 with Narrator.mp4](#) (obsolete) — [Details](#)




Irvan Kurniawan (:sourc7)

Reporter

Updated • 2 years ago

Summary: AddressSanitizer: heap-use-after-free in [mozilla::PresShell::RemoveRefreshObserver] and [@ get] → AddressSanitizer: heap-use-after-free in [mozilla::PresShell::RemoveRefreshObserver]




Daniel Veditz [:dveditz]

Comment 50 • 2 years ago

Given the user interaction required to mostly-reliably trigger this race this is `sec-moderate` in severity

Keywords: [sec-moderate](#)



Eitan Isaacson [:eeejay]

Assignee

Comment 51 • 2 years ago


Comment on [attachment 9223266](#) [\[details\]](#)

[Bug 1709976](#) - Remove selection listeners from shutting down PresShell. r?Jamie

Security Approval Request

- **How easily could an exploit be constructed based on the patch?:** Not at all. This fixes an issue well before any potential UAF.
- **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?:**
- **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?:** Since this patch has no test coverage, and since this is a hard to reproduce UAF I propose that this shouldn't be backported.
- **How likely is this patch to cause regressions; how much testing does it need?:**

[Attachment #9223266](#) - Flags: sec-approval?




Julien Cristau [:jcristau]

Comment 52 • 2 years ago

The sec-moderate rating means you can land without sec-approval.

Flags: needinfo?(eitan)



Irvan Kurniawan (:sourc7)


Reporter

Comment 53 • 2 years ago

(In reply to Eitan Isaacson [:eeejay] from [comment #54](#))

- **If not, how different, hard to create, and risky will they be?:** Since this patch has no test coverage, and since this is a hard to reproduce UAF I propose that this shouldn't be backported.

Gladly! I found new test case that gets triggered by pressing ctrl+shift+m then just with 1-click which is very easy to reproduce. I'll attach the new testcase in a moment.

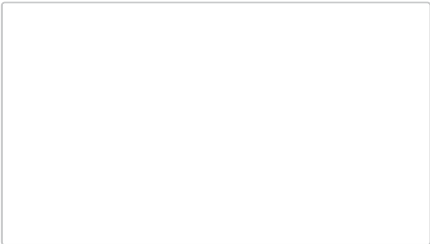


Irvan Kurniawan (:sourc7)

Reporter

Comment 54 • 2 years ago

Attached file [launcher.bundle.html](#) — [Details](#)



Hereby I attach the new testcase "launcher.bundle.html" which triggers UAF more easily and reliably (works every time).
When in responsive design mode then one click inside RDM viewport it will crash with use-after-free.

Steps to reproduce

e10s enabled:

1. Visit attached launcher.bundle.html
2. Click "Launch Testcase"
3. After switched to new window, press `Ctrl + Shift + M` to switch to Responsive Design Mode
4. Click duration section on `<audio>` element or click anywhere inside RDM viewport
5. The tab will crash with use-after-free

e10s disabled:

1. Visit attached testcase.main.html
2. Press `Ctrl + Shift + M` to switch to Responsive Design Mode
3. Click duration section on `<audio>` element or click anywhere inside RDM viewport
4. The tab will crash with use-after-free

[Attachment #9221225](#) - Attachment is obsolete: true
[Attachment #9223279](#) - Attachment is obsolete: true

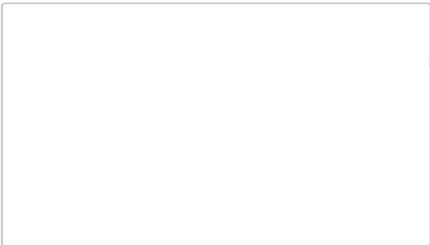


Irvan Kurniawan (:sourc7)
Comment 55 • 2 years ago

Reporter



Attached file [testcase.main.html](#) — Details



[Attachment #9221226](#) - Attachment is obsolete: true
[Attachment #9221227](#) - Attachment is obsolete: true



Irvan Kurniawan (:sourc7)
Comment 56 • 2 years ago

Reporter



Sorry I forgot to mention [on new STR above](#), it require screen reader e.g. Microsoft Narrator, NVDA (on Windows 10), or Orca (on Linux) to activate
Accessibility Activated: `true` on `about:support` in order to reproduce the UAF.

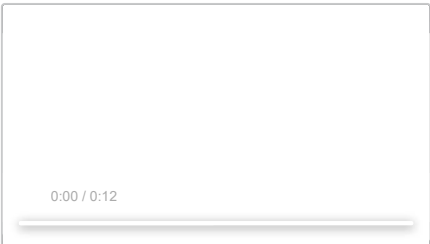


Irvan Kurniawan (:sourc7)
Comment 57 • 2 years ago

Reporter



Attached video [Firefox - UAF in DocAccessibleShutdown with one-click - Windows 10 using Microsoft Narrator.mp4](#) — Details



[Attachment #9221231](#) - Attachment is obsolete: true
[Attachment #9223280](#) - Attachment is obsolete: true



Daniel Veditz [:dveditz]
Updated • 2 years ago










Crash Signature: [`@ nsRefreshDriver::RemoveRefreshObserver`]




Daniel Veditz [:dveditz]
Updated • 2 years ago



Crash Signature: [@ nsRefreshDriver::RemoveRefreshObserver] → [@ nsRefreshDriver::RemoveRefreshObserver] [@ mozilla::detail::MutexImpl::lock] [@ mozilla::PresShell::RemoveRefreshObserver]		
<div> <div>  <div> Daniel Veditz [dveditz] Updated • 2 years ago </div> </div> <div> <div> <div>Type: task → defect</div> <div> status-firefox88: --- → wontfix status-firefox89: --- → wontfix status-firefox90: --- → affected status-firefox91: --- → affected status-firefox-esr78: --- → affected tracking-firefox90: --- → + tracking-firefox91: --- → + tracking-firefox-esr78: --- → 91+ </div> </div> </div> </div>		
<div> <div>  <div> Daniel Veditz [dveditz] Comment 58 • 2 years ago </div> </div> <div> <ul style="list-style-type: none"> Which older supported branches are affected by this flaw?: If not all supported branches, which bug introduced the flaw?: None <p>Presumably all branches are affected based on those answers, and I did reproduce a crash on ESR and Release.</p> <ul style="list-style-type: none"> Do you have backports for the affected branches?: No If not, how different, hard to create, and risky will they be?: Since this patch has no test coverage, and since this is a hard to reproduce UAF I propose that this shouldn't be backported. <p>The code being patched is identical to what's on ESR-78 and Release so it should apply easily, though I can't say whether other relevant context changed around it. It was not hard to reproduce the crash with Irvan's latest testcases. They were very consistent, though less obviously UAF crashes in a release build. The ESR-78 crash behavior was different from the others -- it hung up my browser (partially responsive but mostly dead) and I had to force-kill it from the task manager. The ultimately reported crash had the same signature though -- presumably sent from the crashing tab before the browser became unresponsive as a whole.</p> <p>ESR-78: bp-a4946ab8-9039-468c-b580-d10880210602 Release (88.0.1): bp-cc018aad-9d65-4b2d-84d5-729300210602 Nightly: bp-a3a23889-d9f0-4294-b0d3-29c240210602</p> <p>It would be easy to verify the fix if this is backported, but I don't know about regressions.</p> <ul style="list-style-type: none"> How likely is this patch to cause regressions; how much testing does it need?: <p>This is an important question to get your opinion on.</p> </div> </div>		
status-firefox89 : wontfix → affected tracking-firefox-esr78 : 91+ → ---		
<div> <div>  <div> Eitan Isaacson [eeejay] Assignee Comment 59 • 2 years ago </div> </div> <div> <p>I think the chance that this patch causes a regression is very small but isn't 0. Maybe we can uplift to release, and wait a bit before ESR?</p> <p>Flags: needinfo?(eitan)</p> </div> </div>		
<div> <div>  <div> Tom Ritter [tjr] Comment 60 • 2 years ago </div> </div> <div> <p>Comment on attachment 9223266 [details]</p> <p>bug 4709976 - Remove selection listeners from shutting down PresShell. r?Jamie</p> <p>Approved to land and uplift. I'm not sure about delaying the ESR patch one release; we try to avoid doing that.</p> <p>Attachment #9223266 - Flags: see-approval Attachment #9223266 - Flags: sec-approval+ Attachment #9223266 - Flags: approval-mozilla-esr78+ Attachment #9223266 - Flags: approval-mozilla-beta+</p> </div> </div>		
<div> <div>  <div> Sebastian Hengst [aryx] (needinfo me if it's about an intermittent or bailout) Comment 61 • 2 years ago </div> </div> <div> <p>Remove selection listeners from shutting down PresShell. r=Jamie</p> <p>https://hg.mozilla.org/integration/autoland/rev/65d3494db521b3d15112c092d0647a2899808381 https://hg.mozilla.org/mozilla-central/rev/65d3494db521</p> <p>Group: dom-core-security → core-security-release Status: ASSIGNED → RESOLVED Closed: 2 years ago status-firefox91: affected → fixed Resolution: --- → FIXED Target Milestone: --- → 91 Branch</p> </div> </div>		
<div> <div>  <div> Ryan VanderMeulen [RyanVM] Updated • 2 years ago </div> </div> <div> status-firefox89: affected → wontfix tracking-firefox-esr78: --- → 90+ </div> </div>		
<div> <div>  <div> Julien Cristau [jcristau] Comment 62 • 2 years ago </div> </div> <div> <div>uplift</div> <p>https://hg.mozilla.org/releases/mozilla-beta/rev/81a6b75e5224</p> <p>status-firefox90: affected → fixed</p> </div> </div>		

I still able to reproduce this in Firefox 89.0.2 (64-bit) on Windows 10 (video attached), make sure to click the duration section (0:00 / 0:00) (not the seek bar section) or the white page section as on attached video. I hope this helps!

Flags: ~~needsinfo~~(~~usable~~)


**Rares Doghi**

Comment 68 • 2 years ago

—

Thank you Irvan, I was able to reproduce the issue in older builds and Verify the fix in ESR 78.12.0esr, Beta 90.0b12 and our latest Nightly build on Windows 10 using Narrator and Ubuntu 20.04 using Screen Reader (orca)


Status: RESOLVED → VERIFIED
[status-firefox90: fixed](#) → [verified](#)
[status-firefox91: fixed](#) → [verified](#)
[status-firefox-esr78: fixed](#) → [verified](#)
Flags: ~~needsinfo~~ → [verified](#)

**Tom Ritter [:tjr]**

Updated • 2 years ago

—


Whiteboard: [reporter-external] [client-bounty-form] [verif?][sec-survey] → [reporter-external] [client-bounty-form] [verif?][sec-survey][adv-main90+]

**Tom Ritter [:tjr]**

Updated • 2 years ago

—


Whiteboard: [reporter-external] [client-bounty-form] [verif?][sec-survey][adv-main90+] → [reporter-external] [client-bounty-form] [verif?][sec-survey][adv-main90+][adv-esr78.12+]

**Tom Ritter [:tjr]**

Comment 69 • 2 years ago

—

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


**Tom Ritter [:tjr]**

Comment 70 • 2 years ago

—

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


Attachment #9230315 - Attachment is obsolete: true

**Tom Ritter [:tjr]**

Updated • 2 years ago

—

Alias: CVE-2021-29970

**Daniel Veditz [:dveditz]**

Updated • 1 year ago

—

Group: ~~core-security-release~~

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

Top ↑