

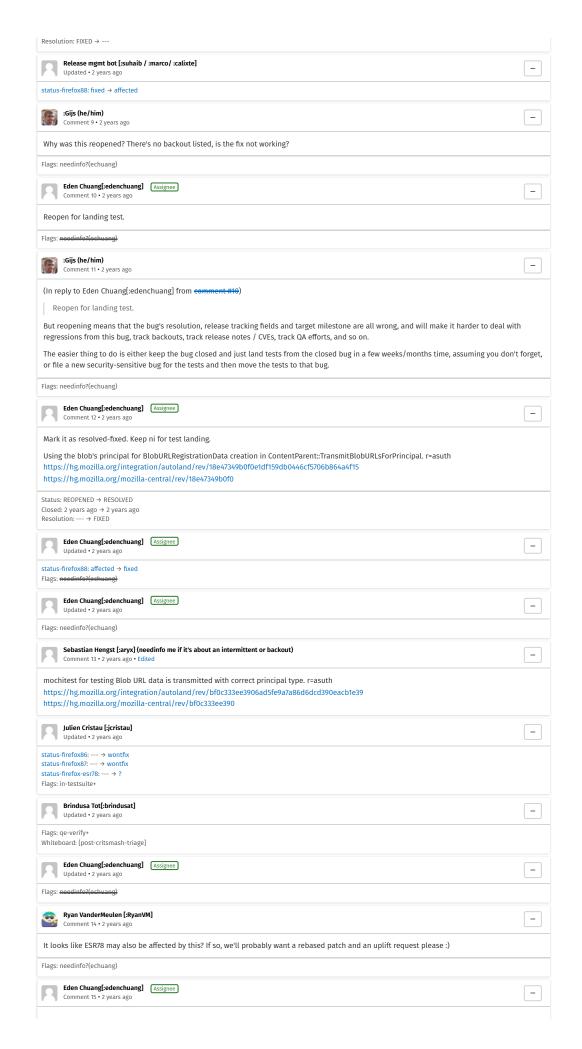
- 1. The code in ContentParent::TransmitBlobURLsForPrincipal includes the wrong principal in the BlobURLRegistrationData to be sent to the content process. Specifically the registration is created with the principal aPrincipal, which is the principal passed to the function, rather than the principal aBlobPrincipal which was used to register the blob. (https://searchfox.org/mozilla-central/rev/400614dec36182ba6dObec2a18044c47df960a1f/dom/ipc/ContentParent.cpp#6123,6137). This means that the wrong principal will be transmitted in cases where these principals don't match.
- 2. The code uses aPrincipal->Subsumes(aBlobPrincipal) instead of ->Equals, meaning it will pass if aPrincipal is an expanded or system principal. This combines with the first issue to mean that if aPrincipal is the system principal, a registration will be transmitted with the system principal for all registered Blob URLs (https://searchfox.org/mozilla-central/rev/400614dec36182ba6d0bec2a18044c47df960a1f/dom/ipc/ContentParent.cpp#6126)
- 3. When bug 1626573 was fixed, logic was added to various codepaths which can cause URLs to load from the parent process to try to force blob URLs to be registered in the content process before the load completes. Unfortunately, this code passes triggeringPrincipal as the principal to use (https://searchfox.org/mozilla-

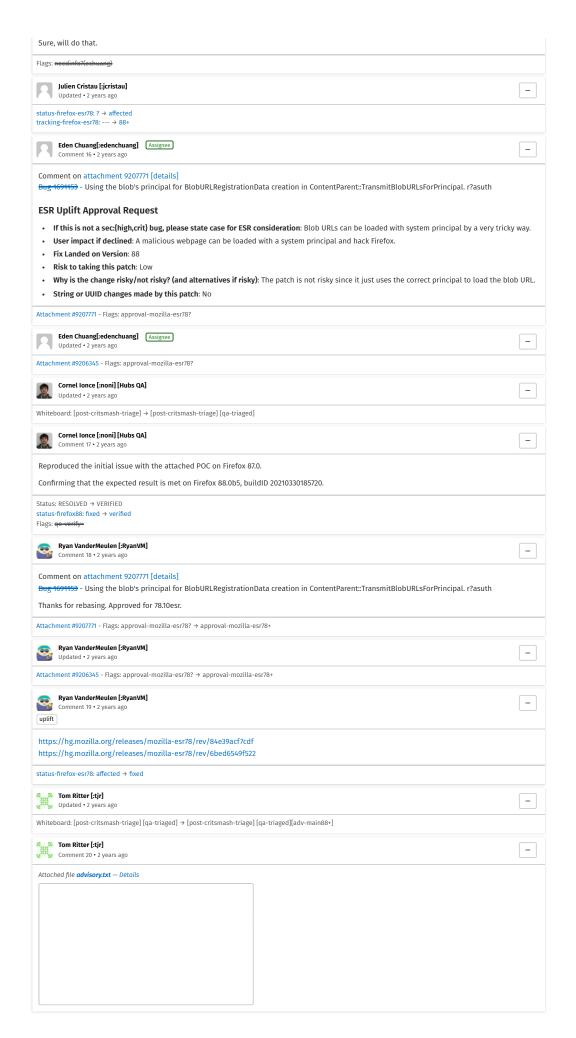
central/rev/400614dec36182ba6d0bec2a18044c47df960a1f/docshell/base/BrowsingContext.cpp#1847-1848, https://searchfox.org/mozilla-central/rev/400614dec36182ba6d0bec2a18044c47df960a1f/netwerk/ipc/DocumentLoadListener.cpp#1835), which is the system principal in the case where the load is triggered by browser UI (e.g. through pasting a URL in the address bar). This combines with the previous two bugs to cause us to register blob URLs with the wrong (potentially system) principal.

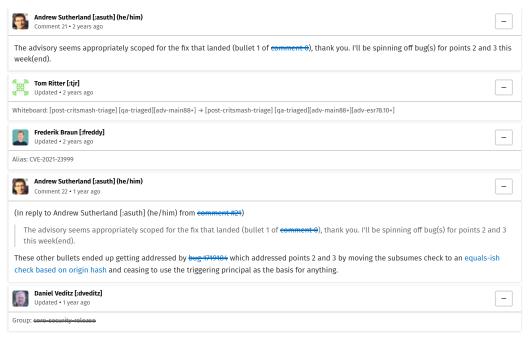
## **Potential Fix**

We should fix all 3 of these core issues in order to make sure we transmit the correct blob URLs with the correct principals. The first issue can be fixed by passing aBlobPrincipal instead of aPrincipal to the various callsites in the function. The second can be fixed by using ->Equals instead of ->Subsumes.

The third bug should be fixed by using the actual principal of the relevant blob URL rather than the triggering principal. This should be doable by removing the aPrincipal argument from TransmitBlobDataIfBlobURL, and instead implementing it as: nsCOMPtr<nsIPrincipal> blobPrincipal; if (BlobURLProtocolHandler::GetBlobURLPrincipal(aURI, getter\_AddRefs(blobPrincipal))) { TransmitBlobURLsForPrincipal(blobPrincipal); **Exploitability** While this is a serious issue, we are fortunate that it is somewhat tricky to get full system-principal permissions with it. The user has to load the URL in question in a different process than the original page, and the triggering principal must be the system principal (which hopefully requires direct user intervention), such as the user pasting the blob URL into their address bar. Andrew Sutherland [:asuth] (he/him) \_ Comment 1 • 2 years ago I'm going to try and dig into understanding and enumerating the motivating use cases for the blob URL transmission to make sure we don't regress any current blob URL use cases without intent in addressing points 2 and 3. The highest priority is addressing point 1 of the Likely Causes, as that's the core of the problem and principal confusion, so if 2/3 look like they'll take longer, we should likely focus on addressing point 1. Status: NEW → ASSIGNED Priority: -- → P1 Andrew Sutherland [:asuth] (he/him) \_ Comment 2 • 2 years ago Nika provided a pernosco trace on Friday as well: https://pernos.co/debug/U3Q4v\_1wxjdpsFmKDDiyYw/index.html Andrew McCreight [:mccr8] Updated • 2 years ago Group: 👄 Tom Tung [:tt, :ttung] -I am going to set the severity to S2 because the workaround doesn't seem to exist but it requires getting system principal which makes this harder Andrew, please feel free to adjust the severity if you have different thoughts on this. Severity: -- → S2 Daniel Veditz [:dveditz] \_ Updated • 2 years ago Keywords: csectype-priy-escalation, sec-moderate Frederik Braun [:freddv] Comment 5 • 2 years as Attached file Bug 1691153 - mochitest for testing Blob URL data is transmitted with correct principal type. r=asuth — Details Frederik Braun [:freddv] \_ @asuth: started writing a test because of the dupe. Feel free to steal, when you get to this bug . Updated • 2 years ago Assignee: bugmail → echuang \_ Comment 7 • 2 years ago Attached file Bug 1691153 - Using the blob's principal for BlobURLRegistrationData creation in ContentParent::TransmitBlobURLsForPrincipal. r?asuth — Details **Phabricator Automation** -Updated • 2 years ago Attachment #9206345 - Attachment description: Bug 1691153 - wip test → Bug 1691153 - mochitest for testing Blob URL data is transmitted with correct principal type. r=asuth Sebastian Hengst [:aryx] (needinfo me if it's about an intermittent or backout) -Comment 8 • 2 years ago Using the blob's principal for BlobURLRegistrationData creation in ContentParent::TransmitBlobURLsForPrincipal. r=asuth https://hg.mozilla.org/integration/autoland/rev/18e47349b0f0e1df159db0446cf5706b864a4f15 https://hg.mozilla.org/mozilla-central/rev/18e47349b0f0 Group: dom-core-security  $\rightarrow$  core-security-release Status: ASSIGNED → RESOLVED Closed: 2 years ago status-firefox88: --- → fixed Resolution: --- → FIXED Target Milestone: --- → 88 Branch \_ Updated • 2 years ago Status: RESOLVED → REOPENED







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

Top ↑