Starred by 4 users

Owner:	t...@google.com
CC:	kouhei@chromium.org jdeblasio@chromium.org
Status:	Fixed *(Closed)*
Components:	Blink>Storage>FileSystem
Modified:	Oct 29, 2021
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Android, Windows, Chrome, Mac, Fuchsia, Lacros
Pri:	1
Type:	Bug-Security

Hotlist-Merge-Review
Security_Impact-Stable
Hotlist-Merge-Approved
Security_Severity-High
allpublic
reward-inprocess
reward-20000
CVE_description-submitted
M-91
Target-91
external_security_report
Target-93
merge-merged-4430
merge-merged-90
FoundIn-91
merge-merged-4472
merge-merged-91
LTS-Merged-90
LTS-Security-90
merge-merged-4515
merge-merged-92
merge-merged-4577
merge-merged-93
LTS-Size-Small
LTS-Complexity-Trivial
Release-1-M92
CVE-2021-30591

Issue 1229298: Security: Chrome: UAF in BindFileUtilitiesHost

Reported by soulc...@gmail.com on Wed, Jul 14, 2021, 2:50 PM EDT

Code

This template is ONLY for reporting security bugs. If you are reporting a Download Protection Bypass bug, please use the "Security - Download Protection" template. For all other reports, please use a different template.

Please READ THIS FAQ before filing a bug: https://chromium.googlesource.com/chromium/src/+/HEAD/docs/security/faq.md

Please see the following link for instructions on filing security bugs: https://www.chromium.org/Home/chromium-security/reporting-security-bugs

Reports may be eligible for reward payments under the Chrome VRP: http://g.co/ChromeBugRewards

NOTE: Security bugs are normally made public once a fix has been widely deployed.

VULNERABILITY DETAILS

There is a race-condition bug in browser process. Here is the code from |PopulateServiceWorkerBinders| in https://source.chromium.org/chromium/chromium/src/+/main:content/browser/browser_interface_binders.cc;l=1227:

```
 map->Add<blink::mojom::FileUtilitiesHost>(
     base::BindRepeating(&BindFileUtilitiesHost, host),
     base::ThreadPool::CreateSequencedTaskRunner(
         {base::MayBlock(), base::TaskPriority::USER_VISIBLE}));
```

It transfer the |host| raw pointer to function |BindFileUtilitiesHost|, but the callback will runs on a separate task runner. On the other hand, we can free the |ServiceWorkerHost::host| object on the browser main thread.

So in the function |BindFileUtilitiesHost| it uses the dangling pointer |host| to call |host->worker_process_id()| to create a |FileUtilitiesHostImpl| object. In the futher exploit, we can fake the process_id to create a more power |FileUtilitiesHostImpl| object, and use this interface to read files out of sandbox.

how to reproduce:
$python ./copy_mojo_js_bindings.py /path/to/chrome/.../out/Asan/gen
$out/Asan/chrome --enable-blink-features=MojoJS 'http://localhost:8000/1.html'

Note that this is *not* a renderer bug; it's a browser process bug that's reachable from the renderer.

Please note that this bug may be can trigger with normal render process(not compromised) if we can find some web API to create many FileUtilitiesHost mojo interface. In theory, create ONLY ONE FileUtilitiesHost can trigger the bug too, but it is hard to win the race.

VERSION
Chrome Version: compile the ASAN version chromium with latest code

REPRODUCTION CASE
Please include a demonstration of the security bug, such as an attached HTML or binary file that reproduces the bug when loaded in Chrome. PLEASE make the file as small as possible and remove any content not required to demonstrate the bug, or any personal or confidential information.

Please attach files directly, not in zip or other archive formats, and if you've created a demonstration site please also attach the files needed to reproduce the demonstration locally.

FOR CRASHES, PLEASE INCLUDE THE FOLLOWING ADDITIONAL INFORMATION
Type of crash: browser
Crash State:
==196032==ERROR: AddressSanitizer: heap-use-after-free on address 0x12046446efc0 at pc 0x7ffed2cbea0b bp 0x00d9047fee20 sp 0x00d9047fee68
READ of size 4 at 0x12046446efc0 thread T32
    #0 0x7ffed2cbea0a in content::ServiceWorkerHost::worker_process_id content\browser\service_worker\service_worker_host.h:59
    #1 0x7ffed2cbea0a in content::internal::`anonymous namespace'::BindFileUtilitiesHost content\browser\browser_interface_binders.cc:370:1
    #2 0x7ffed2cc682d in base::internal::FunctorTraits<void (*)(const content::ServiceWorkerHost *, mojo::PendingReceiver<blink::mojom::FileUtilitiesHost>),void>::Invoke base\bind_internal.h:404
    #3 0x7ffed2cc682d in base::internal::InvokeHelper<0,void>::MakeItSo base\bind_internal.h:648
    #4 0x7ffed2cc682d in base::internal::Invoker<base::internal::BindState<void (*)(const content::ServiceWorkerHost *, mojo::PendingReceiver<blink::mojom::FileUtilitiesHost>),content::ServiceWorkerHost *>,void (mojo::PendingReceiver<blink::mojom::FileUtilitiesHost>)>::RunImpl base\bind_internal.h:721
    #5 0x7ffed2cc682d in base::internal::Invoker<struct base::internal::BindState<void (__cdecl *)(class content::ServiceWorkerHost const *, class mojo::PendingReceiver<class blink::mojom::FileUtilitiesHost>), class content::ServiceWorkerHost *>, (class mojo::PendingReceiver<class blink::mojom::FileUtilitiesHost>)>::Run(class base::internal::BindStateBase *, class mojo::PendingReceiver<class blink::mojom::FileUtilitiesHost> &&) base\bind_internal.h:703:12
    #6 0x7ffed2cc207f in base::RepeatingCallback<void (mojo::PendingReceiver<blink::mojom::AppCacheBackend>)>::Run base\callback.h:167
    #7 0x7ffed2cc207f in mojo::internal::BinderContextTraits<void>::BindGenericReceiver<class blink::mojom::WebUsbService>(class base::RepeatingCallback<(class mojo::PendingReceiver<class blink::mojom::WebUsbService>)> const &, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>) mojo\public\cpp\bindings\lib\binder_map_internal.h:69:12
    #8 0x7ffed2cc2282 in base::internal::FunctorTraits<void (*)(const base::RepeatingCallback<void (mojo::PendingReceiver<blink::mojom::AppCacheBackend>)> &, mojo::ScopedHandleBase<mojo::MessagePipeHandle>),void>::Invoke base\bind_internal.h:404
    #9 0x7ffed2cc2282 in base::internal::InvokeHelper<0,void>::MakeItSo base\bind_internal.h:648
    #10 0x7ffed2cc2282 in base::internal::Invoker<base::internal::BindState<void (*)(const base::RepeatingCallback<void (mojo::PendingReceiver<blink::mojom::AppCacheBackend>)> &, mojo::ScopedHandleBase<mojo::MessagePipeHandle>),base::RepeatingCallback<void (mojo::PendingReceiver<blink::mojom::AppCacheBackend>)> >,void (mojo::ScopedHandleBase<mojo::MessagePipeHandle>)>::RunImpl base\bind_internal.h:721
    #11 0x7ffed2cc2282 in base::internal::Invoker<struct base::internal::BindState<void (__cdecl *)(class base::RepeatingCallback<(class mojo::PendingReceiver<class blink::mojom::WebUsbService>)> const &, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>), class base::RepeatingCallback<void __cdecl(class mojo::PendingReceiver<class blink::mojom::WebUsbService>)>>, (class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>)>::Run(class base::internal::BindStateBase *, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle> &&) base\bind_internal.h:703:12
    #12 0x7ffed3500800 in base::RepeatingCallback<void (mojo::ScopedHandleBase<mojo::MessagePipeHandle>)>::Run base\callback.h:167
    #13 0x7ffed3500800 in mojo::internal::GenericCallbackBinderWithContext<void>::RunCallback(class base::RepeatingCallback<(class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>)> const &, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>) mojo\public\cpp\bindings\lib\binder_map_internal.h:121:14
    #14 0x7ffed3500a35 in base::internal::FunctorTraits<void (*)(const base::RepeatingCallback<void (mojo::ScopedHandleBase<mojo::MessagePipeHandle>)> &, mojo::ScopedHandleBase<mojo::MessagePipeHandle>),void>::Invoke base\bind_internal.h:404
    #15 0x7ffed3500a35 in base::internal::InvokeHelper<0,void>::MakeItSo base\bind_internal.h:648
    #16 0x7ffed3500a35 in base::internal::Invoker<base::internal::BindState<void (*)(const base::RepeatingCallback<void (mojo::ScopedHandleBase<mojo::MessagePipeHandle>)> &, mojo::ScopedHandleBase<mojo::MessagePipeHandle>),base::RepeatingCallback<void (mojo::ScopedHandleBase<mojo::MessagePipeHandle>)>,mojo::ScopedHandleBase<mojo::MessagePipeHandle> >,void ()>::RunImpl base\bind_internal.h:721
    #17 0x7ffed3500a35 in base::internal::Invoker<struct base::internal::BindState<void (__cdecl *)(class base::RepeatingCallback<(class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>)> const &, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>), class base::RepeatingCallback<void __cdecl(class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>)>, class mojo::ScopedHandleBase<class mojo::MessagePipeHandle>>, (void)>::RunOnce(class base::internal::BindStateBase *) base\bind_internal.h:690:12
    #18 0x7ffed965b4ba in base::OnceCallback<void ()>::Run base\callback.h:99
    #19 0x7ffed965b4ba in base::TaskAnnotator::RunTask(char const *, struct base::PendingTask *) base\task\common\task_annotator.cc:178:33
    #20 0x7ffedee2a456 in base::internal::TaskTracker::RunSkipOnShutdown(struct base::internal::Task *) base\task\thread_pool\task_tracker.cc:664:19
    #21 0x7ffedee293ce in base::internal::TaskTracker::RunTaskWithShutdownBehavior base\task\thread_pool\task_tracker.cc:682
    #22 0x7ffedee293ce in base::internal::TaskTracker::RunTask(struct base::internal::Task, class base::internal::TaskSource *, class base::TaskTraits const &) base\task\thread_pool\task_tracker.cc:525:5
    #23 0x7ffedee2871a in base::internal::TaskTracker::RunAndPopNextTask(class base::internal::RegisteredTaskSource) base\task\thread_pool\task_tracker.cc:432:5
    #24 0x7ffee2d4a053 in base::internal::WorkerThread::RunWorker(void) base\task\thread_pool\worker_thread.cc:367:34
    #25 0x7ffee2d491cb in base::internal::WorkerThread::RunPooledWorker(void) base\task\thread_pool\worker_thread.cc:262:3
    #26 0x7ffed9729d1f in base::`anonymous namespace'::ThreadFunc base\threading\platform_thread_win.cc:121:13
    #27 0x7ff6c6a22077 in __asan::AsanThread::ThreadStart(unsigned __int64) C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_thread.cpp:278
    #28 0x7fff61647033 (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #29 0x7fff61ba2650 (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

0x12046446efc0 is located 0 bytes inside of 296-byte region [0x12046446efc0,0x12046446f0e8)
freed by thread T0 here:
    #0 0x7ff6c6a182fb in free C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_malloc_win.cpp:82
    #1 0x7ffed3a49764 in std::__1::unique_ptr<content::ServiceWorkerHost,std::__1::default_delete<content::ServiceWorkerHost> >::~unique_ptr buildtools\third_party\libc++\trunk\include__memory\unique_ptr.h:269
    #2 0x7ffed3a49764 in content::ServiceWorkerVersion::~ServiceWorkerVersion(void) content\browser\service_worker\service_worker_version.cc:292:1
    #3 0x7ffed3a677af in content::ServiceWorkerVersion::`scalar deleting dtor'(unsigned int) content\browser\service_worker\service_worker_version.h:273
    #4 0x7ffed39ce5e7 in scoped_refptr<content::ServiceWorkerVersion>::Release base\memory\scoped_refptr.h:322
    #5 0x7ffed39ce5e7 in scoped_refptr<content::ServiceWorkerVersion>::~scoped_refptr base\memory\scoped_refptr.h:224
    #6 0x7ffed39ce5e7 in content::ServiceWorkerObjectHost::~ServiceWorkerObjectHost(void) content\browser\service_worker\service_worker_object_host.cc:214:1
    #7 0x7ffed39d1359 in content::ServiceWorkerObjectHost::`scalar deleting dtor'(unsigned int) content\browser\service_worker\service_worker_object_host.cc:211:53
    #8 0x7ffed392cd8d in std::__1::default_delete<content::ServiceWorkerObjectHost>::operator() buildtools\third_party\libc++\trunk\include__memory\unique_ptr.h:54
    #9 0x7ffed392cd8d in std::__1::unique_ptr<content::ServiceWorkerObjectHost,std::__1::default_delete<content::ServiceWorkerObjectHost> >::reset buildtools\third_party\libc++\trunk\include__memory\unique_ptr.h:315
    #10 0x7ffed392cd8d in std::__1::unique_ptr<content::ServiceWorkerObjectHost,std::__1::default_delete<content::ServiceWorkerObjectHost> >::~unique_ptr buildtools\third_party\libc++\trunk\include__memory\unique_ptr.h:269
    #11 0x7ffed392cd8d in content::ServiceWorkerContainerHost::RemoveServiceWorkerObjectHost(__int64) content\browser\service_worker\service_worker_container_host.cc:736:1
    #12 0x7ffed9ab9f0b in base::RepeatingCallback<void ()>::Run base\callback.h:166
    #13 0x7ffed9ab9f0b in mojo::ReceiverSetState::OnDisconnect(unsigned __int64, unsigned int, class std::__1::basic_string<char, struct std::__1::char_traits<char>, class std::__1::allocator<char>> const &) mojo\public\cpp\bindings\receiver_set.cc:141:25
    #14 0x7ffed9a9860c in base::OnceCallback<void (unsigned int, const std::__1::string &)>::Run base\callback.h:99
    #15 0x7ffed9a9860c in mojo::InterfaceEndpointClient::NotifyError(class absl::optional<struct mojo::DisconnectReason> const &) mojo\public\cpp\bindings\lib\interface_endpoint_client.cc:683:45
    #16 0x7ffed9aae199 in mojo::internal::MultiplexRouter::ProcessNotifyErrorTask(struct mojo::internal::MultiplexRouter::Task *, enum mojo::internal::MultiplexRouter::ClientCallBehavior, class base::SequencedTaskRunner *) mojo\public\cpp\bindings\lib\multiplex_router.cc:1019:13
    #17 0x7ffed9aa8756 in mojo::internal::MultiplexRouter::ProcessTasks(enum mojo::internal::MultiplexRouter::ClientCallBehavior, class base::SequencedTaskRunner *) mojo\public\cpp\bindings\lib\multiplex_router.cc:932:15
    #18 0x7ffed9aab575 in mojo::internal::MultiplexRouter::Accept(class mojo::Message *) mojo\public\cpp\bindings\lib\multiplex_router.cc:744:5
    #19 0x7ffedc2aff72 in mojo::MessageDispatcher::Accept(class mojo::Message *) mojo\public\cpp\bindings\lib\message_dispatcher.cc:43:19
    #20 0x7ffed9a8f380 in mojo::Connector::DispatchMessageW(class mojo::Message) mojo\public\cpp\bindings\lib\connector.cc:548:49

#21 0x7ffed9a90b22 in mojo::Connector::ReadAllAvailableMessages(void) mojo\public\cpp\bindings\lib\connector.cc:606:14
#22 0x7ffed965b4ba in base::OnceCallback<void ()>::Run base\callback.h:99
#23 0x7ffed965b4ba in base::TaskAnnotator::RunTask(char const *, struct base::PendingTask *) base\task\common\task_annotator.cc:178:33
#24 0x7ffedbe510d0 in base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWorkImpl(class base::sequence_manager::LazyNow *)
base\task\sequence_manager\thread_controller_with_message_pump_impl.cc:360:23
#25 0x7ffedbe50732 in base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork(void)
base\task\sequence_manager\thread_controller_with_message_pump_impl.cc:260:36
#26 0x7ffed97083a6 in base::MessagePumpForUI::DoRunLoop(void) base\message_loop\message_pump_win.cc:220:67
#27 0x7ffed97064ff in base::MessagePumpWin::Run(class base::MessagePump::Delegate *) base\message_loop\message_pump_win.cc:78:3
#28 0x7ffedbe525c0 in base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run(bool, class base::TimeDelta)
base\task\sequence_manager\thread_controller_with_message_pump_impl.cc:467:12
#29 0x7ffed95dd113 in base::RunLoop::Run(class base::Location const &) base\run_loop.cc:134:14
#30 0x7ffed2cceb21 in content::BrowserMainLoop::RunMainMessageLoop(void) content\browser\browser_main_loop.cc:999:18
#31 0x7ffed2cd4129 in content::BrowserMainRunnerImpl::Run(void) content\browser\browser_main_runner_impl.cc:152:15
#32 0x7ffed2cc80ee in content::BrowserMain(struct content::MainFunctionParams const &) content\browser\browser_main.cc:47:28
#33 0x7ffed9361518 in content::RunBrowserProcessMain(struct content::MainFunctionParams const &, class content::ContentMainDelegate *)
content\app\content_main_runner_impl.cc:595:10
#34 0x7ffed9363edc in content::ContentMainRunnerImpl::RunBrowser(struct content::MainFunctionParams &, bool) content\app\content_main_runner_impl.cc:1084:10
#35 0x7ffed9363056 in content::ContentMainRunnerImpl::Run(bool) content\app\content_main_runner_impl.cc:953:12
#36 0x7ffed9360332 in content::RunContentProcess(struct content::ContentMainParams const &, class content::ContentMainRunner *)
content\app\content_main.cc:386:36

previously allocated by thread T0 here:
#0 0x7ff6c6a183fb in malloc C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_malloc_win.cpp:98
#1 0x7ffeebd1b06e in operator new(unsigned __int64) d:\agent_work\4\s\src\vctools\crt\vcstartup\src\heap\new_scalar.cpp:35
#2 0x7ffed3a63b7a in std::__1::make_unique buildtools\third_party\libc++\trunk\include__memory\unique_ptr.h:725
#3 0x7ffed3a63b7a in content::ServiceWorkerVersion::StartWorkerInternal(void) content\browser\service_worker\service_worker_version.cc:1887:18
#4 0x7ffed3a4e044 in content::ServiceWorkerVersion::DidEnsureLiveRegistrationForStartWorker(enum content::ServiceWorkerMetrics::EventType, enum
content::ServiceWorkerVersion::Status, bool, class base::OnceCallback<(enum blink::ServiceWorkerStatusCode)>, enum blink::ServiceWorkerStatusCode, class
scoped_refptr<class content::ServiceWorkerRegistration>) content\browser\service_worker\service_worker_version.cc:1865:5
#5 0x7ffed3a692b0 in base::internal::FunctorTraits<void (content::ServiceWorkerVersion::*)(content::ServiceWorkerMetrics::EventType,
content::ServiceWorkerVersion::Status, bool, base::OnceCallback<void (blink::ServiceWorkerStatusCode)>, blink::ServiceWorkerStatusCode,
scoped_refptr<content::ServiceWorkerRegistration>),void>::Invoke base\bind_internal.h:509
#6 0x7ffed3a692b0 in base::internal::InvokeHelper<1,void>::MakeItSo base\bind_internal.h:668
#7 0x7ffed3a692b0 in base::internal::Invoker<base::internal::BindState<void (content::ServiceWorkerVersion::*)(content::ServiceWorkerMetrics::EventType,
content::ServiceWorkerVersion::Status, bool, base::OnceCallback<void (blink::ServiceWorkerStatusCode)>, blink::ServiceWorkerStatusCode,
scoped_refptr<content::ServiceWorkerRegistration>),base::WeakPtr<content::ServiceWorkerVersion>,content::ServiceWorkerMetrics::EventType,content::ServiceWorkerVers
ion::Status,bool,base::OnceCallback<void (blink::ServiceWorkerStatusCode)> >,void (blink::ServiceWorkerStatusCode,
scoped_refptr<content::ServiceWorkerRegistration>)>::RunImpl base\bind_internal.h:721
#8 0x7ffed3a692b0 in base::internal::Invoker<struct base::internal::BindState<void (__cdecl content::ServiceWorkerVersion::*)(enum
content::ServiceWorkerMetrics::EventType, enum content::ServiceWorkerVersion::Status, bool, class base::OnceCallback<(enum blink::ServiceWorkerStatusCode)>, enum
blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>), class base::WeakPtr<class content::ServiceWorkerVersion>, enum
content::ServiceWorkerMetrics::EventType, enum content::ServiceWorkerVersion::Status, bool, class base::OnceCallback<void __cdecl(enum
blink::ServiceWorkerStatusCode)>>, (enum blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>)>::RunOnce(class
base::internal::BindStateBase *, enum blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration> &&) base\bind_internal.h:690:12
#9 0x7ffed39696bf in base::OnceCallback<(enum blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>)>::Run(enum
blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>) && base\callback.h:98:12
#10 0x7ffed3a1212d in content::`anonymous namespace'::CompleteFindNow content\browser\service_worker\service_worker_registry.cc:65:23
#11 0x7ffed39ff8fe in content::ServiceWorkerRegistry::FindRegistrationForIdInternal(__int64, class absl::optional<class blink::StorageKey> const &, class
base::OnceCallback<(enum blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>)>)
content\browser\service_worker\service_worker_registry.cc:749:5
#12 0x7ffed39ff4ca in content::ServiceWorkerRegistry::FindRegistrationForId(__int64, class blink::StorageKey const &, class base::OnceCallback<(enum
blink::ServiceWorkerStatusCode, class scoped_refptr<class content::ServiceWorkerRegistration>)>) content\browser\service_worker\service_worker_registry.cc:244:3
#13 0x7ffed3a4bfc8 in content::ServiceWorkerVersion::StartWorker(enum content::ServiceWorkerMetrics::EventType, class base::OnceCallback<(enum
blink::ServiceWorkerStatusCode)>) content\browser\service_worker\service_worker_version.cc:470:25
#14 0x7ffed39e13f5 in content::ServiceWorkerRegisterJob::StartWorkerForUpdate(class scoped_refptr<class content::ServiceWorkerVersion>)
content\browser\service_worker\service_worker_register_job.cc:534:18
#15 0x7ffed3a14e1f in base::OnceCallback<(class scoped_refptr<class content::ServiceWorkerVersion>)>::Run(class scoped_refptr<class
content::ServiceWorkerVersion>) && base\callback.h:98:12
#16 0x7ffed39fd399 in content::ServiceWorkerRegistry::DidGetNewVersionId(class scoped_refptr<class content::ServiceWorkerRegistration>, class GURL const &, enum
blink::mojom::ScriptType, class base::OnceCallback<(class scoped_refptr<class content::ServiceWorkerVersion>)>, __int64, class mojo::PendingRemote<class
storage::mojom::ServiceWorkerLiveVersionRef>) content\browser\service_worker\service_worker_registry.cc:1395:23
#17 0x7ffed3a186f2 in base::internal::FunctorTraits<void (__cdecl content::ServiceWorkerRegistry::*)(class scoped_refptr<class content::ServiceWorkerRegistration>,
class GURL const &, enum blink::mojom::ScriptType, class base::OnceCallback<(class scoped_refptr<class content::ServiceWorkerVersion>)>, __int64, class
mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>), void>::Invoke<void (__cdecl content::ServiceWorkerRegistry::*)(class scoped_refptr<class
content::ServiceWorkerRegistration>, class GURL const &, enum blink::mojom::ScriptType, class base::OnceCallback<(class scoped_refptr<class
content::ServiceWorkerVersion>)>, __int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>), class base::WeakPtr<class
content::ServiceWorkerRegistry>, class scoped_refptr<class content::ServiceWorkerRegistration>, class GURL, enum blink::mojom::ScriptType, class
base::OnceCallback<void __cdecl(class scoped_refptr<class content::ServiceWorkerVersion>)>, __int64, class mojo::PendingRemote<class
storage::mojom::ServiceWorkerLiveVersionRef>>(void (__cdecl content::ServiceWorkerRegistry::*)(class scoped_refptr<class content::ServiceWorkerRegistration>, class
GURL const &, enum blink::mojom::ScriptType, class base::OnceCallback<(class scoped_refptr<class content::ServiceWorkerVersion>)>, __int64, class
mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>), class base::WeakPtr<class content::ServiceWorkerRegistry> &&, class scoped_refptr<class
content::ServiceWorkerRegistration> &&, class GURL &&, enum blink::mojom::ScriptType &&, class base::OnceCallback<(class scoped_refptr<class
content::ServiceWorkerVersion>)> &&, __int64 &&, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef> &&) base\bind_internal.h:509:12
#18 0x7ffed3a184dc in base::internal::InvokeHelper<1,void>::MakeItSo base\bind_internal.h:668
#19 0x7ffed3a184dc in base::internal::Invoker<base::internal::BindState<void (content::ServiceWorkerRegistry::*)(scoped_refptr<content::ServiceWorkerRegistration>,
const GURL &, blink::mojom::ScriptType, base::OnceCallback<void (scoped_refptr<content::ServiceWorkerVersion>)>, long long,
mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>),base::WeakPtr<content::ServiceWorkerRegistry>,scoped_refptr<content::ServiceWorkerRegistration
>,GURL,blink::mojom::ScriptType,base::OnceCallback<void (scoped_refptr<content::ServiceWorkerVersion>)> >,void (long long,
mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>)>::RunImpl base\bind_internal.h:721
#20 0x7ffed3a184dc in base::internal::Invoker<struct base::internal::BindState<void (__cdecl content::ServiceWorkerRegistry::*)(class scoped_refptr<class
content::ServiceWorkerRegistration>, class GURL const &, enum blink::mojom::ScriptType, class base::OnceCallback<(class scoped_refptr<class
content::ServiceWorkerVersion>)>, __int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>), class base::WeakPtr<class
content::ServiceWorkerRegistry>, class scoped_refptr<class content::ServiceWorkerRegistration>, class GURL, enum blink::mojom::ScriptType, class
base::OnceCallback<void __cdecl(class scoped_refptr<class content::ServiceWorkerVersion>)>>, (__int64, class mojo::PendingRemote<class
storage::mojom::ServiceWorkerLiveVersionRef>)>::RunOnce(class base::internal::BindStateBase *, __int64, class mojo::PendingRemote<class
storage::mojom::ServiceWorkerLiveVersionRef> &&) base\bind_internal.h:690:12
#21 0x7ffed3a18e8f in base::OnceCallback<void (long long, mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>)>::Run base\callback.h:99
#22 0x7ffed3a18e8f in content::InflightCallWithInvoker<__int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>>::DidReply(__int64,
class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>) content\browser\service_worker\service_worker_registry.cc:134:31
#23 0x7ffed3a1909f in base::internal::FunctorTraits<void (content::InflightCallWithInvoker<long
long,mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef> >::*)(long long,
mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>),void>::Invoke base\bind_internal.h:509
#24 0x7ffed3a1909f in base::internal::InvokeHelper<0,void>::MakeItSo base\bind_internal.h:648
#25 0x7ffed3a1909f in base::internal::Invoker<base::internal::BindState<void (content::InflightCallWithInvoker<long
long,mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef> >::*)(long long,
mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>),base::internal::UnretainedWrapper<content::InflightCallWithInvoker<long
long,mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef> > > >,void (long long,
mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>)>::RunImpl base\bind_internal.h:721
#26 0x7ffed3a1909f in base::internal::Invoker<struct base::internal::BindState<void (__cdecl content::InflightCallWithInvoker<__int64, class mojo::PendingRemote<class
storage::mojom::ServiceWorkerLiveVersionRef>>::*)(__int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>), class
base::internal::UnretainedWrapper<class content::InflightCallWithInvoker<__int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>>>>,
(__int64, class mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef>)>::RunOnce(class base::internal::BindStateBase *, __int64, class

```
mojo::PendingRemote<class storage::mojom::ServiceWorkerLiveVersionRef> &&) base\bind_internal.h:690:12
    #27 0x7ffed28604be in base::OnceCallback<void (long long, mojo::PendingRemote<storage::mojom::ServiceWorkerLiveVersionRef>)>::Run base\callback.h:99
    #28 0x7ffed28604be in storage::mojom::ServiceWorkerStorageControl_GetNewVersionId_ForwardToCallback::Accept(class mojo::Message *)
out\Debug_ASAN\gen\components\services\storage\public\mojom\service_worker_storage_control.mojom.cc:6423:26
    #29 0x7ffed9a943aa in mojo::InterfaceEndpointClient::HandleValidatedMessage(class mojo::Message *) mojo\public\cpp\bindings\lib\interface_endpoint_client.cc:885:23
    #30 0x7ffedc2aff72 in mojo::MessageDispatcher::Accept(class mojo::Message *) mojo\public\cpp\bindings\lib\message_dispatcher.cc:43:19
    #31 0x7ffed9a97e50 in mojo::InterfaceEndpointClient::HandleIncomingMessage(class mojo::Message *) mojo\public\cpp\bindings\lib\interface_endpoint_client.cc:649:21
    #32 0x7ffed9aac75a in mojo::internal::MultiplexRouter::ProcessIncomingMessage(class mojo::internal::MultiplexRouter::MessageWrapper *, enum
mojo::internal::MultiplexRouter::ClientCallBehavior, class base::SequencedTaskRunner *) mojo\public\cpp\bindings\lib\multiplex_router.cc:1099:42
    #33 0x7ffed9aab4b4 in mojo::internal::MultiplexRouter::Accept(class mojo::Message *) mojo\public\cpp\bindings\lib\multiplex_router.cc:719:7
    #34 0x7ffedc2aff72 in mojo::MessageDispatcher::Accept(class mojo::Message *) mojo\public\cpp\bindings\lib\message_dispatcher.cc:43:19
    #35 0x7ffed9a8f380 in mojo::Connector::DispatchMessageW(class mojo::Message) mojo\public\cpp\bindings\lib\connector.cc:548:49
    #36 0x7ffed9a90b22 in mojo::Connector::ReadAllAvailableMessages(void) mojo\public\cpp\bindings\lib\connector.cc:606:14
    #37 0x7ffed9ae1588 in base::RepeatingCallback<void (unsigned int, const mojo::HandleSignalsState &)>::Run base\callback.h:166
    #38 0x7ffed9ae1588 in mojo::SimpleWatcher::OnHandleReady(int, unsigned int, struct mojo::HandleSignalsState const &)
mojo\public\cpp\system\simple_watcher.cc:278:14
    #39 0x7ffed965b4ba in base::OnceCallback<void ()>::Run base\callback.h:99
    #40 0x7ffed965b4ba in base::TaskAnnotator::RunTask(char const *, struct base::PendingTask *) base\task\common\task_annotator.cc:178:33

Thread T32 created by T30 here:
    #0 0x7ff6c6a22ae2 in __asan_wrap_CreateThread C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_win.cpp:146
    #1 0x7ffed97290fe in base::`anonymous namespace'::CreateThreadInternal base\threading\platform_thread_win.cc:185:7
    #2 0x7ffee2d480d0 in base::internal::WorkerThread::Start(class base::WorkerThreadObserver *) base\task\thread_pool\worker_thread.cc:109:3
    #3 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl::<lambda_2>::operator() base\task\thread_pool\thread_group_impl.cc:187
    #4 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::WorkerContainer::ForEachWorker<class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>>(class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>) base\task\thread_pool\thread_group_impl.cc:153:9
    #5 0x7ffedee4244f in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void) base\task\thread_pool\thread_group_impl.cc:185:23
    #6 0x7ffedee3c011 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushWorkerCreation base\task\thread_pool\thread_group_impl.cc:119
    #7 0x7ffedee3c011 in base::internal::ThreadGroupImpl::WorkerThreadDelegateImpl::GetWork(class base::internal::WorkerThread *)
base\task\thread_pool\thread_group_impl.cc:596:14
    #8 0x7ffee2d49fdc in base::internal::WorkerThread::RunWorker(void) base\task\thread_pool\worker_thread.cc:354:51
    #9 0x7ffee2d491cb in base::internal::WorkerThread::RunPooledWorker(void) base\task\thread_pool\worker_thread.cc:262:3
    #10 0x7ffed9729d1f in base::`anonymous namespace'::ThreadFunc base\threading\platform_thread_win.cc:121:13
    #11 0x7ff6c6a22077 in __asan::AsanThread::ThreadStart(unsigned __int64) C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_thread.cpp:278
    #12 0x7fff61647033  (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #13 0x7fff61ba2650  (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

Thread T30 created by T21 here:
    #0 0x7ff6c6a22ae2 in __asan_wrap_CreateThread C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_win.cpp:146
    #1 0x7ffed97290fe in base::`anonymous namespace'::CreateThreadInternal base\threading\platform_thread_win.cc:185:7
    #2 0x7ffee2d480d0 in base::internal::WorkerThread::Start(class base::WorkerThreadObserver *) base\task\thread_pool\worker_thread.cc:109:3
    #3 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl::<lambda_2>::operator() base\task\thread_pool\thread_group_impl.cc:187
    #4 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::WorkerContainer::ForEachWorker<class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>>(class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>) base\task\thread_pool\thread_group_impl.cc:153:9
    #5 0x7ffedee4244f in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void) base\task\thread_pool\thread_group_impl.cc:185:23
    #6 0x7ffedee3a0ce in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::~ScopedCommandsExecutor(void)
base\task\thread_pool\thread_group_impl.cc:104:31
    #7 0x7ffedee3d977 in base::internal::ThreadGroupImpl::WorkerThreadDelegateImpl::DidProcessTask(class base::internal::RegisteredTaskSource)
base\task\thread_pool\thread_group_impl.cc:673:1
    #8 0x7ffee2d4a0e2 in base::internal::WorkerThread::RunWorker(void) base\task\thread_pool\worker_thread.cc:369:16
    #9 0x7ffee2d491cb in base::internal::WorkerThread::RunPooledWorker(void) base\task\thread_pool\worker_thread.cc:262:3
    #10 0x7ffed9729d1f in base::`anonymous namespace'::ThreadFunc base\threading\platform_thread_win.cc:121:13
    #11 0x7ff6c6a22077 in __asan::AsanThread::ThreadStart(unsigned __int64) C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_thread.cpp:278
    #12 0x7fff61647033  (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #13 0x7fff61ba2650  (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

Thread T21 created by T9 here:
    #0 0x7ff6c6a22ae2 in __asan_wrap_CreateThread C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_win.cpp:146
    #1 0x7ffed97290fe in base::`anonymous namespace'::CreateThreadInternal base\threading\platform_thread_win.cc:185:7
    #2 0x7ffee2d480d0 in base::internal::WorkerThread::Start(class base::WorkerThreadObserver *) base\task\thread_pool\worker_thread.cc:109:3
    #3 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl::<lambda_2>::operator() base\task\thread_pool\thread_group_impl.cc:187
    #4 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::WorkerContainer::ForEachWorker<class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>>(class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>) base\task\thread_pool\thread_group_impl.cc:153:9
    #5 0x7ffedee4244f in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void) base\task\thread_pool\thread_group_impl.cc:185:23
    #6 0x7ffedee3c011 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushWorkerCreation base\task\thread_pool\thread_group_impl.cc:119
    #7 0x7ffedee3c011 in base::internal::ThreadGroupImpl::WorkerThreadDelegateImpl::GetWork(class base::internal::WorkerThread *)
base\task\thread_pool\thread_group_impl.cc:596:14
    #8 0x7ffee2d49fdc in base::internal::WorkerThread::RunWorker(void) base\task\thread_pool\worker_thread.cc:354:51
    #9 0x7ffee2d491cb in base::internal::WorkerThread::RunPooledWorker(void) base\task\thread_pool\worker_thread.cc:262:3
    #10 0x7ffed9729d1f in base::`anonymous namespace'::ThreadFunc base\threading\platform_thread_win.cc:121:13
    #11 0x7ff6c6a22077 in __asan::AsanThread::ThreadStart(unsigned __int64) C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_thread.cpp:278
    #12 0x7fff61647033  (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #13 0x7fff61ba2650  (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

Thread T9 created by T7 here:
    #0 0x7ff6c6a22ae2 in __asan_wrap_CreateThread C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_win.cpp:146
    #1 0x7ffed97290fe in base::`anonymous namespace'::CreateThreadInternal base\threading\platform_thread_win.cc:185:7
    #2 0x7ffee2d480d0 in base::internal::WorkerThread::Start(class base::WorkerThreadObserver *) base\task\thread_pool\worker_thread.cc:109:3
    #3 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl::<lambda_2>::operator() base\task\thread_pool\thread_group_impl.cc:187
    #4 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::WorkerContainer::ForEachWorker<class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>>(class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>) base\task\thread_pool\thread_group_impl.cc:153:9
    #5 0x7ffedee4244f in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void) base\task\thread_pool\thread_group_impl.cc:185:23
    #6 0x7ffedee3c011 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushWorkerCreation base\task\thread_pool\thread_group_impl.cc:119
    #7 0x7ffedee3c011 in base::internal::ThreadGroupImpl::WorkerThreadDelegateImpl::GetWork(class base::internal::WorkerThread *)
base\task\thread_pool\thread_group_impl.cc:596:14
    #8 0x7ffee2d49fdc in base::internal::WorkerThread::RunWorker(void) base\task\thread_pool\worker_thread.cc:354:51
    #9 0x7ffee2d491cb in base::internal::WorkerThread::RunPooledWorker(void) base\task\thread_pool\worker_thread.cc:262:3
    #10 0x7ffed9729d1f in base::`anonymous namespace'::ThreadFunc base\threading\platform_thread_win.cc:121:13
    #11 0x7ff6c6a22077 in __asan::AsanThread::ThreadStart(unsigned __int64) C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_thread.cpp:278
    #12 0x7fff61647033  (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #13 0x7fff61ba2650  (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

Thread T7 created by T0 here:
    #0 0x7ff6c6a22ae2 in __asan_wrap_CreateThread C:\b\s\w\ir\cache\builder\src\third_party\llvm\compiler-rt\lib\asan\asan_win.cpp:146
    #1 0x7ffed97290fe in base::`anonymous namespace'::CreateThreadInternal base\threading\platform_thread_win.cc:185:7
    #2 0x7ffee2d480d0 in base::internal::WorkerThread::Start(class base::WorkerThreadObserver *) base\task\thread_pool\worker_thread.cc:109:3
    #3 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl::<lambda_2>::operator() base\task\thread_pool\thread_group_impl.cc:187
    #4 0x7ffedee42920 in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::WorkerContainer::ForEachWorker<class `private: void __cdecl
base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>>(class `private: void __cdecl
```

base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void)'::`1'::<lambda_2>) base\task\thread_pool\thread_group_impl.cc:153:9
    #5 0x7ffedee4244f in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::FlushImpl(void) base\task\thread_pool\thread_group_impl.cc:185:23
    #6 0x7ffedee3a0ce in base::internal::ThreadGroupImpl::ScopedCommandsExecutor::~ScopedCommandsExecutor(void)
base\task\thread_pool\thread_group_impl.cc:104:31
    #7 0x7ffedee394f5 in base::internal::ThreadGroupImpl::Start(int, int, class base::TimeDelta, class scoped_refptr<class base::SequencedTaskRunner>, class
base::WorkerThreadObserver *, enum base::internal::ThreadGroup::WorkerEnvironment, bool, class absl::optional<class base::TimeDelta>)
base\task\thread_pool\thread_group_impl.cc:425:1
    #8 0x7ffedbe5d569 in base::internal::ThreadPoolImpl::Start(struct base::ThreadPoolInstance::InitParams const &, class base::WorkerThreadObserver *)
base\task\thread_pool\thread_pool_impl.cc:230:11
    #9 0x7ffed3abee6d in content::StartBrowserThreadPool(void) content\browser\startup_helper.cc:95:36
    #10 0x7ffed93636ec in content::ContentMainRunnerImpl::RunBrowser(struct content::MainFunctionParams &, bool) content\app\content_main_runner_impl.cc:1017:7
    #11 0x7ffed9363056 in content::ContentMainRunnerImpl::Run(bool) content\app\content_main_runner_impl.cc:953:12
    #12 0x7ffed9360332 in content::RunContentProcess(struct content::ContentMainParams const &, class content::ContentMainRunner *)
content\app\content_main.cc:386:36
    #13 0x7ffed936091a in content::ContentMain(struct content::ContentMainParams const &) content\app\content_main.cc:412:10
    #14 0x7ffecef7145a in ChromeMain chrome\app\chrome_main.cc:151:12
    #15 0x7ff6c6975bb4 in MainDllLoader::Launch(struct HINSTANCE__*, class base::TimeTicks) chrome\app\main_dll_loader_win.cc:169:12
    #16 0x7ff6c6972be8 in main chrome\app\chrome_exe_main_win.cc:382:20
    #17 0x7ff6c6d6ac6f in invoke_main d:\agent_work\4\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl:78
    #18 0x7ff6c6d6ac6f in __scrt_common_main_seh d:\agent_work\4\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl:288
    #19 0x7fff61647033 (C:\Windows\System32\KERNEL32.DLL+0x180017033)
    #20 0x7fff61ba2650 (C:\Windows\SYSTEM32\ntdll.dll+0x180052650)

SUMMARY: AddressSanitizer: heap-use-after-free content\browser\service_worker\service_worker_host.h:59 in content::ServiceWorkerHost::worker_process_id
Shadow bytes around the buggy address:
  0x0420f0b8dda0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0420f0b8ddb0: fd fd fd fd fd fd fd fd fd fa fa fa fa fa fa fa
  0x0420f0b8ddc0: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0420f0b8ddd0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0420f0b8dde0: fd fd fd fd fd fd fd fd fd fd fa fa fa fa fa fa
=>0x0420f0b8ddf0: fa fa fa fa fa fa fa fa[fd]fd fd fd fd fd fd fd
  0x0420f0b8de00: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0420f0b8de10: fd fd fd fd fd fd fd fd fd fd fd fd fa fa fa fa
  0x0420f0b8de20: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0420f0b8de30: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0420f0b8de40: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable: 00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone: fa
  Freed heap region: fd
  Stack left redzone: f1
  Stack mid redzone: f2
  Stack right redzone: f3
  Stack after return: f5
  Stack use after scope: f8
  Global redzone: f9
  Global init order: f6
  Poisoned by user: f7
  Container overflow: fc
  Array cookie: ac
  Intra object redzone: bb
  ASan internal: fe
  Left alloca redzone: ca
  Right alloca redzone: cb
Client ID (if relevant): [see link above]

CREDIT INFORMATION
Externally reported security bugs may appear in Chrome release notes. If this bug is included, how would you like to be credited?
Reporter credit: SorryMybad(@S0rryMybad) of Kunlun Lab

I also upload the patch.diff for your reference. If you use it, can you add my email address to AUTHORS?

1.html
218 bytes View Download

copy_mojo_js_bindings.py
512 bytes View Download

patch.diff
1.1 KB View Download

sw.js
418 bytes View Download

Comment 1 by sheriffbot on Wed, Jul 14, 2021, 2:53 PM EDT Project Member

Labels: external_security_report

Comment 2 by jdeblasio@chromium.org on Wed, Jul 14, 2021, 5:07 PM EDT Project Member

Status: Assigned (was: Unconfirmed)
Owner: talp@chromium.org
Labels: Security_Severity-High FoundIn-91 OS-Android OS-Chrome OS-Fuchsia OS-Linux OS-Mac OS-Windows OS-Lacros Pri-1
Components: Blink>Storage>FileSystem

Hi talp@: please take a look at this as soon as possible?

Marking this a Severity-High because a web API that can create many FileUtilitiesHost interfaces is only theoretical. However, this certainly seems high risk (and at the high end of Sev-High), and if you can find one, this bug would be Severity-Critical.

Comment 3 by sheriffbot on Wed, Jul 14, 2021, 5:11 PM EDT Project Member

Labels: Security_Impact-Stable

Comment 4 by sheriffbot on Thu, Jul 15, 2021, 9:06 AM EDT Project Member

Labels: M-91 Target-91

Setting milestone and target because of high severity.

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

Comment 5 by soulc...@gmail.com on Mon, Jul 19, 2021, 9:21 AM EDT

ANY UPDATE HERE?

Comment 6 by t...@google.com on Mon, Jul 19, 2021, 9:36 PM EDT Project Member

Owner: jdeblasio@chromium.org
Cc: talp@chromium.org

Apologies for the delay, I was OOO.

While I can't comment on the severity, the analysis and suggested fix seem correct.

> I also upload the patch.diff for your reference. If you use it, can you add my email address to AUTHORS?
I don't know if there's a way to create a commit with a different author than myself (and AFAIK the AUTHORS file is populated automatically based on commit authors). If you'd like to try to create and submit the change yourself, I'm more than happy to help with that.

jdeblasio@, any suggestions on how we should proceed with this?

Comment 7 by jdeblasio@chromium.org on Tue, Jul 20, 2021, 2:58 PM EDT Project Member

Owner: t...@google.com
Cc: -talp@chromium.org jdeblasio@chromium.org

Re: AUTHORS file, I don't think there's any automated tooling. I agree with talp@: if you'd like to send out a change and ask talp@ or I to review it, we'd be happy to add you. See https://chromium.googlesource.com/chromium/src/+/refs/heads/main/docs/contributing.md#initial-git-setup for getting started instructions.

talp@, now that you're back, can you upload the fix if you think it's adequate, so that we aren't postponing this fix any longer than necessary? Thanks!

Comment 8 by t...@google.com on Tue, Jul 20, 2021, 8:56 PM EDT Project Member

Ack, I'll send a CL shortly. I assume we'll want to merge it to M93?

Comment 9 by t...@google.com on Tue, Jul 20, 2021, 9:35 PM EDT Project Member

Cc: kouhei@chromium.org

Comment 10 by t...@google.com on Tue, Jul 20, 2021, 11:50 PM EDT Project Member

Ah, now I remember why I didn't pass the process ID instead of binding the host in the first place - the process may not exist at the time the binder map is populated. See PS1 here[1] for sample failures.

So I'm going to use a WeakPtr instead, hopefully that won't cause any test failures.

Comment 11 by t...@google.com on Tue, Jul 20, 2021, 11:50 PM EDT Project Member

And, of course, I forgot the actual link in the previous comment:
https://chromium-review.googlesource.com/c/chromium/src/+/3041006/1

Comment 12 by Git Watcher on Wed, Jul 21, 2021, 5:12 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/e2123a8e0943b4399814bd530df401ed071b6d0f

commit e2123a8e0943b4399814bd530df401ed071b6d0f
Author: Tal Pressman <talp@chromium.org>
Date: Wed Jul 21 09:11:13 2021

Manually post task to bind FileUtilitiesHost.

The FileUtilitiesHost binder is posted to a separate sequence, and the
ServiceWorkerHost may be destroyed by the time the it runs, causing a
UAF.
This CL changes it so that, when we try to bind a new receiver, the
host's worker_process_id() is obtained first (on the service worker's
core thread) and then a task is posted to do the actual binding on a
USER_VISIBLE task runner.

Credit: This issue was first reported (with analysis) by
soulchen8650@gmail.com.

~~Bug: 1230208~~
Change-Id: I6d5c05a830ba30f6cb98bf2df70a3df3333f3dd9
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3041006
Reviewed-by: Kinuko Yasuda <kinuko@chromium.org>
Reviewed-by: Kouhei Ueno <kouhei@chromium.org>
Commit-Queue: Tal Pressman <talp@google.com>
Cr-Commit-Position: refs/heads/master@{#903832}

[modify] https://crrev.com/e2123a8e0943b4399814bd530df401ed071b6d0f/content/browser/browser_interface_binders.cc

Comment 13 by jdeblasio@chromium.org on Wed, Jul 21, 2021, 1:25 PM EDT Project Member

Re: #8: Yes please. Once this bug is marked Fixed, sheriffbot will likely auto-add the merge request labels if you do not.

Comment 14 by t...@google.com on Wed, Jul 21, 2021, 10:15 PM EDT Project Member

Status: Fixed (was: Assigned)

Thanks. I'm marking as fixed, and will wait for the bot and the questionnaire.

Comment 15 by sheriffbot on Thu, Jul 22, 2021, 12:42 PM EDT Project Member

Labels: reward-topanel

Comment 16 by sheriffbot on Thu, Jul 22, 2021, 1:42 PM EDT Project Member

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

Comment 17 by sheriffbot on Sat, Jul 24, 2021, 9:10 AM EDT Project Member

Labels: Merge-Request-92 Merge-Request-93 Merge-Request-91

Requesting merge to extended stable M91 because latest trunk commit (903832) appears to be after extended stable branch point (870763).

Requesting merge to stable M92 because latest trunk commit (903832) appears to be after stable branch point (885287).

Requesting merge to dev M93 because latest trunk commit (903832) appears to be after dev branch point (902210).

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

Comment 18 by sheriffbot on Sat, Jul 24, 2021, 9:12 AM EDT Project Member

Labels: -Merge-Request-92 Merge-Review-92 Hotlist-Merge-Review

This bug requires manual review: Request affecting a post-stable build
Before a merge request will be considered, the following information is required to be added to this bug:

1. Does your merge fit within the Merge Decision Guidelines?
- Chrome: https://chromium.googlesource.com/chromium/src.git/+/main/docs/process/merge_request.md#when-to-request-a-merge
- Chrome OS: https://goto.google.com/cros-release-branch-merge-guidelines
2. Links to the CLs you are requesting to merge.
3. Has the change landed and been verified on ToT?
4. Does this change need to be merged into other active release branches (M-1, M+1)?
5. Why are these changes required in this milestone after branch?
6. Is this a new feature?
7. If it is a new feature, is it behind a flag using finch?

Chrome OS Only:
8. Was the change reviewed and approved by the Eng Prod Representative? See Eng Prod ownership by component: http://go/cros-engprodcomponents

Please contact the milestone owner if you have questions.
Owners: govind@(Android), benmason@(iOS), dgagnon@(ChromeOS), srinivassista@(Desktop)

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

Comment 19 by pbommana@google.com on Sat, Jul 24, 2021, 9:28 AM EDT Project Member

Labels: Target-93

Comment 20 by sheriffbot on Sun, Jul 25, 2021, 9:13 AM EDT Project Member

Labels: -Merge-Request-93 Hotlist-Merge-Approved Merge-Approved-93

Your change meets the bar and is auto-approved for M93. Please go ahead and merge the CL to branch 4577 (refs/branch-heads/4577) manually. Please contact milestone owner if you have questions.
Merge instructions: https://www.chromium.org/developers/how-tos/drover
Owners: benmason@(Android), govind@(iOS), geohsu@(ChromeOS), pbommana@(Desktop)

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

Comment 21 by pbommana@google.com on Sun, Jul 25, 2021, 9:15 AM EDT Project Member

Your change has been approved for M93. Please go ahead and merge the CL to branch 4577 manually asap so that it would be part of this week's Dev/Beta release.

Comment 22 by Git Watcher on Mon, Jul 26, 2021, 11:06 PM EDT Project Member

Labels: -merge-approved-93 merge-merged-4577 merge-merged-93

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/a498cfe6a2294c5e8023a31000605d62538b5c29

commit a498cfe6a2294c5e8023a31000605d62538b5c29
Author: Tal Pressman <talp@chromium.org>
Date: Tue Jul 27 03:05:48 2021

Manually post task to bind FileUtilitiesHost.

This is a merge of crrev.com/c/3041006 to the M93 branch.

The FileUtilitiesHost binder is posted to a separate sequence, and the
ServiceWorkerHost may be destroyed by the time the it runs, causing a
UAF.
This CL changes it so that, when we try to bind a new receiver, the
host's worker_process_id() is obtained first (on the service worker's
core thread) and then a task is posted to do the actual binding on a
USER_VISIBLE task runner.

Credit: This issue was first reported (with analysis) by
soulchen8650@gmail.com.

(cherry picked from commit e2123a8e0943b4399814bd530df401ed071b6d0f)

~~Bug: 1220208~~
Change-Id: I6d5c05a830ba30f6cb98bf2df70a3df3333f3dd9
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3041006
Reviewed-by: Kinuko Yasuda <kinuko@chromium.org>
Reviewed-by: Kouhei Ueno <kouhei@chromium.org>
Commit-Queue: Tal Pressman <talp@google.com>
Cr-Original-Commit-Position: refs/heads/master@{#903832}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3052875
Reviewed-by: Prudhvi Kumar Bommana <pbommana@google.com>
Cr-Commit-Position: refs/branch-heads/4577@{#204}
Cr-Branched-From: 761ddde228655e313424edec06497d0c56b0f3c4-refs/heads/master@{#902210}

[modify] https://crrev.com/a498cfe6a2294c5e8023a31000605d62538b5c29/content/browser/browser_interface_binders.cc

Comment 23 by t...@google.com on Tue, Jul 27, 2021, 12:47 AM EDT Project Member

The fix was merged to the M93 branch, but I'm a little confused by all the comments and labels added by sheriffbot.

jdeblasio@
I answered the questionnaire to the best of my ability below. Could you please take a look (esp. questions 1, 4 and 5) and advise on how to proceed?

==========
1. Does your merge fit within the Merge Decision Guidelines?
The CL was automatically approved and merged to M93. For M92(and possibly 91?) it is a post-stable fix, so I'll defer to jdeblasio@ regarding criticality.

2. Links to the CLs you are requesting to merge.
crrev.com/c/3041006 (crrev.com/c/3052875 is the merged M93 CL).

3. Has the change landed and been verified on ToT?
Yes.

4. Does this change need to be merged into other active release branches (M-1, M+1)?
5. Why are these changes required in this milestone after branch?
I'll defer to jdeblasio@ on these questions.

6. Is this a new feature?
No, this is a security fix for code that landed in M91.

7. If it is a new feature, is it behind a flag using finch?
Neither the fix nor the original code it fixes are behind a flag.

Comment 24 by jdeblasio@chromium.org on Tue, Jul 27, 2021, 1:03 PM EDT Project Member

Re: questions 1, 4 and 5, this is a high-severity security vulnerability present in M91/92.

Comment 25 by amyressler@google.com on Tue, Jul 27, 2021, 4:26 PM EDT Project Member

Labels: -Merge-Request-91 -Merge-Review-92 Merge-Approved-92 Merge-Approved-91

Sorry for all the flurry of comments and labels by Sheriffbot. Your CL landed in time that it could be auto-approved for merge to M93; however, since this bug was found it M91, the bot has also helpfully kicked off merge review for M92 and M91.

Please go ahead and merge to M92, branch 4515, by Thursday so this fix for this high severity issue can be a part of the stable refresh release next week.
Also, at your earliest convenience, please also merge to branch 4472, as M91 is now the extended stable release as we move toward the 4W stable release cycle.

Comment 26 by t...@google.com on Tue, Jul 27, 2021, 7:24 PM EDT Project Member

Thanks for the clarification! I'll start the CLs to merge to M91 and M92.

Comment 27 by Git Watcher on Tue, Jul 27, 2021, 11:01 PM EDT Project Member

Labels: -merge-approved-92 merge-merged-4515 merge-merged-92

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/0f1eaba52798718df865c9cafd7fdafb880343df

commit 0f1eaba52798718df865c9cafd7fdafb880343df
Author: Tal Pressman <talp@chromium.org>
Date: Wed Jul 28 03:00:15 2021

Manually post task to bind FileUtilitiesHost.

This is a merge of crrev.com/c/3041006 to the M92 branch.

The FileUtilitiesHost binder is posted to a separate sequence, and the
ServiceWorkerHost may be destroyed by the time the it runs, causing a
UAF.
This CL changes it so that, when we try to bind a new receiver, the
host's worker_process_id() is obtained first (on the service worker's
core thread) and then a task is posted to do the actual binding on a
USER_VISIBLE task runner.

Credit: This issue was first reported (with analysis) by
soulchen8650@gmail.com.

(cherry picked from commit e2123a8e0943b4399814bd530df401ed071b6d0f)

~~Bug: 1239208~~
Change-Id: I6d5c05a830ba30f6cb98bf2df70a3df3333f3dd9
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3041006
Reviewed-by: Kinuko Yasuda <kinuko@chromium.org>
Reviewed-by: Kouhei Ueno <kouhei@chromium.org>
Commit-Queue: Tal Pressman <talp@google.com>
Cr-Original-Commit-Position: refs/heads/master@{#903832}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3056313
Auto-Submit: Tal Pressman <talp@google.com>
Commit-Queue: Kinuko Yasuda <kinuko@chromium.org>
Cr-Commit-Position: refs/branch-heads/4515@{#1856}
Cr-Branched-From: 488fc70865ddaa05324ac00a54a6eb783b4bc41c-refs/heads/master@{#885287}

[modify] https://crrev.com/0f1eaba52798718df865c9cafd7fdafb880343df/content/browser/browser_interface_binders.cc

Comment 28 by Git Watcher on Tue, Jul 27, 2021, 11:02 PM EDT Project Member

Labels: -merge-approved-91 merge-merged-4472 merge-merged-91

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/f024e59f574ee3ca4297d35810e4c96ebd223173

commit f024e59f574ee3ca4297d35810e4c96ebd223173
Author: Tal Pressman <talp@chromium.org>
Date: Wed Jul 28 03:01:32 2021

Manually post task to bind FileUtilitiesHost.

This is a merge of crrev.com/c/3041006 to the M91 branch.

The FileUtilitiesHost binder is posted to a separate sequence, and the
ServiceWorkerHost may be destroyed by the time the it runs, causing a
UAF.
This CL changes it so that, when we try to bind a new receiver, the
host's worker_process_id() is obtained first (on the service worker's
core thread) and then a task is posted to do the actual binding on a
USER_VISIBLE task runner.

Credit: This issue was first reported (with analysis) by
soulchen8650@gmail.com.

(cherry picked from commit e2123a8e0943b4399814bd530df401ed071b6d0f)

~~Bug: 1239208~~
Change-Id: I6d5c05a830ba30f6cb98bf2df70a3df3333f3dd9
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3041006
Reviewed-by: Kinuko Yasuda <kinuko@chromium.org>
Reviewed-by: Kouhei Ueno <kouhei@chromium.org>
Commit-Queue: Tal Pressman <talp@google.com>
Cr-Original-Commit-Position: refs/heads/master@{#903832}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3058010
Auto-Submit: Tal Pressman <talp@google.com>
Commit-Queue: Kinuko Yasuda <kinuko@chromium.org>
Cr-Commit-Position: refs/branch-heads/4472@{#1584}
Cr-Branched-From: 3d60439cfb36485e76a1c5bb7f513d3721b20da1-refs/heads/master@{#870763}

[modify] https://crrev.com/f024e59f574ee3ca4297d35810e4c96ebd223173/content/browser/browser_interface_binders.cc

Comment 29 by amyressler@google.com on Wed, Jul 28, 2021, 4:51 PM EDT Project Member

Labels: -reward-topanel reward-unpaid reward-20000

*** Boilerplate reminders! ***
Please do NOT publicly disclose details until a fix has been released to all our users. Early public disclosure may cancel the provisional reward. Also, please be considerate about disclosure when the bug affects a core library that may be used by other products. Please do NOT share this information with third parties who are not directly involved in fixing the bug. Doing so may cancel the provisional reward. Please be honest if you have already disclosed anything publicly or to third parties. Lastly, we understand that some of you are not interested in money. We offer the option to donate your reward to an eligible charity. If you prefer this option, let us know and we will also match your donation - subject to our discretion. Any rewards that are unclaimed after 12 months will be donated to a charity of our choosing.

Please contact security-vrp@chromium.org with any questions.
****

Comment 30 by amyressler@google.com on Wed, Jul 28, 2021, 5:12 PM EDT Project Member

Congratulations, SorryMybad! The VRP Panel has decided to award you $20,000 for this report. Nice work!

Comment 31 by amyressler@google.com on Thu, Jul 29, 2021, 5:48 PM EDT Project Member

Labels: -reward-unpaid reward-inprocess

Comment 32 by amyressler@chromium.org on Mon, Aug 2, 2021, 10:31 AM EDT Project Member

Labels: Release-1-M92

Comment 33 by amyressler@google.com on Mon, Aug 2, 2021, 10:57 AM EDT Project Member

Labels: CVE-2021-30591 CVE_description-missing

Comment 34 by voit@google.com on Thu, Aug 5, 2021, 1:23 AM EDT Project Member

Labels: LTS-Security-90 LTS-Merge-Request-90 LTS-Size-Small LTS-Complexity-Trivial

Comment 35 by gianluca@google.com on Thu, Aug 5, 2021, 6:20 AM EDT Project Member

Labels: -LTS-Merge-Request-90 LTS-Merge-Approved-90

Comment 36 by Git Watcher on Tue, Aug 10, 2021, 2:37 AM EDT Project Member

Labels: merge-merged-4430 merge-merged-90

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/995064c730323afab2ab92fd1fef50f7d41b19f4

commit 995064c730323afab2ab92fd1fef50f7d41b19f4
Author: Tal Pressman <talp@chromium.org>
Date: Tue Aug 10 06:36:26 2021

[M90-LTS] Manually post task to bind FileUtilitiesHost.

The FileUtilitiesHost binder is posted to a separate sequence, and the ServiceWorkerHost may be destroyed by the time the it runs, causing a UAF.
This CL changes it so that, when we try to bind a new receiver, the host's worker_process_id() is obtained first (on the service worker's core thread) and then a task is posted to do the actual binding on a USER_VISIBLE task runner.

Credit: This issue was first reported (with analysis) by soulchen8650@gmail.com.

(cherry picked from commit e2123a8e0943b4399814bd530df401ed071b6d0f)

~~Bug: 1220208~~
Change-Id: I6d5c05a830ba30f6cb98bf2df70a3df3333f3dd9
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3041006
Commit-Queue: Tal Pressman <talp@google.com>
Cr-Original-Commit-Position: refs/heads/master@{#903832}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3071365
Reviewed-by: Achuith Bhandarkar <achuith@chromium.org>
Commit-Queue: Zakhar Voit <voit@google.com>
Owners-Override: Achuith Bhandarkar <achuith@chromium.org>
Cr-Commit-Position: refs/branch-heads/4430@{#1564}
Cr-Branched-From: e5ce7dc4f7518237b3d9bb93cccca35d25216cbe-refs/heads/master@{#857950}

[modify] https://crrev.com/995064c730323afab2ab92fd1fef50f7d41b19f4/content/browser/browser_interface_binders.cc

Comment 37 by voit@google.com on Thu, Aug 12, 2021, 3:07 AM EDT Project Member

Labels: -LTS-Merge-Approved-90 LTS-Merged-90

Comment 38 by amyressler@google.com on Thu, Aug 26, 2021, 1:09 PM EDT Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 39 by sheriffbot on Fri, Oct 29, 2021, 1:30 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