☆ Starred by 2 users

Owner:	sidereal@google.com **OOO until the new year**
CC:	rzanoni@google.com aashay@chromium.org dgagnon@chromium.org
Status:	Fixed *(Closed)*
Components:	UI>Shell
Modified:	Jul 29, 2022
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Chrome
Pri:	1
Type:	Bug-Security

M-100
Security_Severity-High
allpublic
reward-inprocess
Via-Wizard-Security
CVE_description-submitted
external_security_report
FoundIn-91
reward-7000
Target-100
Security_Impact-Extended
merge-merged-4664
LTS-Merge-Merged-96
merge-merged-4896
merge-merged-100
merge-merged-4951
merge-merged-101
Release-2-M100
CVE-2022-1311

Issue 1310717: Use-after-Free on crostini::CrostiniExportImport::OpenFileDialog

Reported by etern...@gmail.com on Mon, Mar 28, 2022, 5:19 AM EDT

Code

UserAgent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36

Steps to reproduce the problem:
1. Refer to the instructions for simple patching of the code
2. Start chromium in chromeos
3. open chrome://crostini-upgrader and click Upgrade, then close chrome://crostini-upgrader.

What is the expected behavior?

What went wrong?
# Use-after-Free on crostini::CrostiniExportImport::OpenFileDialog

## Root Cause and some notes

[0] The `web_contents` is posted to a separate sequence

[1] `web_contents` may be destroyed in UI by the time the it runs, causing a UAF in OnBackupPathChecked(finial in OpenFileDialog) callback.

The pattern of this vulnerability is similar to this issue.
https://bugs.chromium.org/p/chromium/issues/detail?id=1233975

```
void CrostiniUpgrader::Backup(const ContainerId& container_id,
                              bool show_file_chooser,
                              content::WebContents* web_contents) {
  if (show_file_chooser) {
    CrostiniExportImport::GetForProfile(profile_)->ExportContainer(
        container_id, web_contents, MakeFactory());
    return;
  }
  base::FilePath default_path =
      CrostiniExportImport::GetForProfile(profile_)->GetDefaultBackupPath();
  base::ThreadPool::PostTaskAndReplyWithResult(
      FROM_HERE, {base::MayBlock()},
      base::BindOnce(&base::PathExists, default_path),
      base::BindOnce(&CrostiniUpgrader::OnBackupPathChecked,
                     weak_ptr_factory_.GetWeakPtr(), container_id, web_contents,
                     default_path)); //-----> posttask reply here [0]
}

void CrostiniUpgrader::OnBackupPathChecked(const ContainerId& container_id,
                                           content::WebContents* web_contents,

                                           base::FilePath path,
                                           bool path_exists) {
  if (path_exists) {
```

```
  if (path_exists) {
    CrostiniExportImport::GetForProfile(profile_)->ExportContainer(//--->[1]
        container_id, web_contents, MakeFactory());
  } else {
    CrostiniExportImport::GetForProfile(profile_)->ExportContainer(
        container_id, path, MakeFactory());
  }
}

void CrostiniExportImport::ExportContainer(ContainerId container_id,
                                           content::WebContents* web_contents,
                                           OnceTrackerFactory tracker_factory) {
  OpenFileDialog(
      NewOperationData(ExportImportType::EXPORT, std::move(container_id),
                       std::move(tracker_factory)),
      web_contents);//--->[1]
}

void CrostiniExportImport::OpenFileDialog(OperationData* operation_data,
                                          content::WebContents* web_contents) {
  if (!crostini::CrostiniFeatures::Get()->IsExportImportUIAllowed(profile_)) {
    return;
  }

  ui::SelectFileDialog::Type file_selector_mode;
  unsigned title = 0;
  base::FilePath default_path;
  ui::SelectFileDialog::FileTypeInfo file_types;
  file_types.allowed_paths = ui::SelectFileDialog::FileTypeInfo::NATIVE_PATH;
  file_types.extensions = {{"tini", "tar.gz", "tgz"}};

  switch (operation_data->type) {
    case ExportImportType::EXPORT:
      file_selector_mode = ui::SelectFileDialog::SELECT_SAVEAS_FILE;
      title = IDS_SETTINGS_CROSTINI_EXPORT;
      default_path = GetDefaultBackupPath();
      break;
    case ExportImportType::IMPORT:
      file_selector_mode = ui::SelectFileDialog::SELECT_OPEN_FILE,
      title = IDS_SETTINGS_CROSTINI_IMPORT;
      default_path = file_manager::util::GetMyFilesFolderForProfile(profile_);
      break;
  }

  select_folder_dialog_ = ui::SelectFileDialog::Create(
      this, std::make_unique<ChromeSelectFilePolicy>(web_contents));
  select_folder_dialog_->SelectFile(
      file_selector_mode, l10n_util::GetStringUTF16(title), default_path,
      &file_types, 0, base::FilePath::StringType(),
      web_contents->GetTopLevelNativeWindow(),//--------->[1] use here!
      static_cast<void*>(operation_data));

}
```

[2] The root cause of this vulnerability is very clear, but to quick trigger this vulnerability, we need to patch some codes.
1. First add sleep(10) at the beginning of the base::PathExists function.
Since this is a base function, in order not to cause side effects, I added a Test function with a flag

```
void CrostiniUpgrader::Backup(const ContainerId& container_id,
                              bool show_file_chooser,
                              content::WebContents* web_contents) {
  if (show_file_chooser) {
    CrostiniExportImport::GetForProfile(profile_)->ExportContainer(
        container_id, web_contents, MakeFactory());
    return;
  }
  base::FilePath default_path =
      CrostiniExportImport::GetForProfile(profile_)->GetDefaultBackupPath();
  base::ThreadPool::PostTaskAndReplyWithResult(
      FROM_HERE, {base::MayBlock()},
-      base::BindOnce(&base::PathExists, default_path),
+      base::BindOnce(&base::PathExistsForTest, default_path, true),
}

bool PathExistsForTest(const FilePath& path, bool test = true) {
  LOG(ERROR) << "sakura in PathExists" << test << std::endl;
  sleep(10);
  ScopedBlockingCall scoped_blocking_call(FROM_HERE, BlockingType::MAY_BLOCK);
#if BUILDFLAG(IS_ANDROID)
  if (path.IsContentUri()) {
    return ContentUriExists(path);
  }
#endif
  return access(path.value().c_str(), F_OK) == 0;
}
```

2. Since my ChromeOS is compiled on Linux, some functions are missing, but it does not affect the real scene. In order to pass some checks, I patched the following code

```
void CrostiniUpgrader::OnBackupPathChecked(const ContainerId& container_id,
                                           content::WebContents* web_contents,
                                           base::FilePath path,
                                           bool path_exists) {
  // if (path_exists) {
    LOG(ERROR) << "sakura in OnBackupPathChecked" << std::endl;
    CrostiniExportImport::GetForProfile(profile_)->ExportContainer(
        container_id, web_contents, MakeFactory());
  // } else {
  //   LOG(ERROR) << "sakura in OnBackupPathChecked3" << std::endl;
  //   CrostiniExportImport::GetForProfile(profile_)->ExportContainer(
  //       container_id, path, MakeFactory());
  // }
}


void CrostiniExportImport::OpenFileDialog(OperationData* operation_data,
                                          content::WebContents* web_contents) {
```

```
                                    content::WebContents* web_contents) {
  LOG(ERROR) << "sakura in CrostiniExportImport::OpenFileDialog" << std::endl;
  // if (!crostini::CrostiniFeatures::Get()->IsExportImportUIAllowed(profile_)) {
  //   return;
  // }
  LOG(ERROR) << "sakura in CrostiniExportImport::OpenFileDialog2" << std::endl;
  ui::SelectFileDialog::Type file_selector_mode;
  unsigned title = 0;
  base::FilePath default_path;
  ui::SelectFileDialog::FileTypeInfo file_types;
  file_types.allowed_paths = ui::SelectFileDialog::FileTypeInfo::NATIVE_PATH;
  file_types.extensions = {{"tini", "tar.gz", "tgz"}};
```

[0]
https://source.chromium.org/chromium/chromium/src/+/main:chrome/browser/ash/crostini/crostini_upgrader.cc;l=215;drc=8a6efc8a06137639ea50513b0ba62932e9698038

[1]
https://source.chromium.org/chromium/chromium/src/+/main:chrome/browser/ash/crostini/crostini_upgrader.cc;l=225;drc=8a6efc8a06137639ea50513b0ba62932e9698038

https://source.chromium.org/chromium/chromium/src/+/main:chrome/browser/ash/crostini/crostini_export_import.cc;l=144;drc=8a6efc8a06137639ea50513b0ba62932e9698038

https://source.chromium.org/chromium/chromium/src/+/main:chrome/browser/ash/crostini/crostini_export_import.cc;l=198;drc=8a6efc8a06137639ea50513b0ba62932e9698038

## ASAN LOG

```
=================================================================
==519913==ERROR: AddressSanitizer: heap-use-after-free on address 0x61f000221280 at pc 0x555bdf7306cc bp
0x7ffd0ca2b710 sp 0x7ffd0ca2b708
READ of size 8 at 0x61f000221280 thread T0 (chrome)
2022-03-28T08:34:10.411529Z ERROR chrome[519913:519937]: [object_proxy.cc(623)] Failed to call method:
org.chromium.debugd.GetPerfOutputFd: object_path= /org/chromium/debugd: org.freedesktop.DBus.Error.ServiceUnknown:
The name org.chromium.debugd was not provided by any .service files
    #0 0x555bdf7306cb in crostini::CrostiniExportImport::OpenFileDialog(crostini::CrostiniExportImport::OperationData*,
content::WebContents*) chrome/browser/ash/crostini/crostini_export_import.cc:200:21
    #1 0x555bdf7308b5 in crostini::CrostiniExportImport::ExportContainer(crostini::ContainerId, content::WebContents*,
base::OnceCallback<std::__Cr::unique_ptr<crostini::CrostiniExportImportStatusTracker,
std::__Cr::default_delete<crostini::CrostiniExportImportStatusTracker> > (crostini::ExportImportType, base::FilePath)>)
chrome/browser/ash/crostini/crostini_export_import.cc:145:3
    #2 0x555bdf807b3c in crostini::CrostiniUpgrader::OnBackupPathChecked(crostini::ContainerId const&,
content::WebContents*, base::FilePath, bool) chrome/browser/ash/crostini/crostini_upgrader.cc:234:52
    #3 0x555bdf80cf5d in void base::internal::FunctorTraits<void (crostini::CrostiniUpgrader::*)(crostini::ContainerId const&,
content::WebContents*, base::FilePath, bool), void>::Invoke<void (crostini::CrostiniUpgrader::*)(crostini::ContainerId const&,
content::WebContents*, base::FilePath, bool), base::WeakPtr<crostini::CrostiniUpgrader>, crostini::ContainerId,
content::WebContents*, base::FilePath, bool>(void (crostini::CrostiniUpgrader::*)(crostini::ContainerId const&,
content::WebContents*, base::FilePath, bool), base::WeakPtr<crostini::CrostiniUpgrader>&&, crostini::ContainerId&&,
content::WebContents*&&, base::FilePath&&, bool&&) base/bind_internal.h:542:12

    #4 0x555bdf80cd8b in MakeItSo<void (crostini::CrostiniUpgrader::*)(const crostini::ContainerId &, content::WebContents
*, base::FilePath, bool), base::WeakPtr<crostini::CrostiniUpgrader>, crostini::ContainerId, content::WebContents *,
base::FilePath, bool> base/bind_internal.h:726:5
```

```
base::FilePath, bool> base/bind_internal.h:726:5
    #5 0x555bdf80cd8b in RunImpl<void (crostini::CrostiniUpgrader::*)(const crostini::ContainerId &, content::WebContents *,
base::FilePath, bool), std::__Cr::tuple<base::WeakPtr<crostini::CrostiniUpgrader>, crostini::ContainerId,
base::internal::UnretainedWrapper<content::WebContents>, base::FilePath>, 0UL, 1UL, 2UL, 3UL>
base/bind_internal.h:779:12
    #6 0x555bdf80cd8b in base::internal::Invoker<base::internal::BindState<void (crostini::CrostiniUpgrader::*)
(crostini::ContainerId const&, content::WebContents*, base::FilePath, bool), base::WeakPtr<crostini::CrostiniUpgrader>,
crostini::ContainerId, base::internal::UnretainedWrapper<content::WebContents>, base::FilePath>, void
(bool)>::RunOnce(base::internal::BindStateBase*, bool) base/bind_internal.h:748:12
    #7 0x555bdb30a56e in base::OnceCallback<void (bool)>::Run(bool) && base/callback.h:142:12
    #8 0x555bdb3099f5 in void base::internal::ReplyAdapter<bool, bool>(base::OnceCallback<void (bool)>,
std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> >*) base/task/post_task_and_reply_with_result_internal.h:30:23
    #9 0x555bdb30a177 in Invoke<void (*)(base::OnceCallback<void (bool)>, std::__Cr::unique_ptr<bool,
std::__Cr::default_delete<bool> > *), base::OnceCallback<void (bool)>, std::__Cr::unique_ptr<bool,
std::__Cr::default_delete<bool> > *> base/bind_internal.h:437:12
    #10 0x555bdb30a177 in MakeItSo<void (*)(base::OnceCallback<void (bool)>, std::__Cr::unique_ptr<bool,
std::__Cr::default_delete<bool> > *), base::OnceCallback<void (bool)>, std::__Cr::unique_ptr<bool,
std::__Cr::default_delete<bool> > *> base/bind_internal.h:706:12
    #11 0x555bdb30a177 in RunImpl<void (*)(base::OnceCallback<void (bool)>, std::__Cr::unique_ptr<bool,
std::__Cr::default_delete<bool> > *), std::__Cr::tuple<base::OnceCallback<void (bool)>,
base::internal::OwnedWrapper<std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> >,
std::__Cr::default_delete<std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> > > > >, 0UL, 1UL>
base/bind_internal.h:779:12
    #12 0x555bdb30a177 in base::internal::Invoker<base::internal::BindState<void (*)(base::OnceCallback<void (bool)>,
std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> >*), base::OnceCallback<void (bool)>,
base::internal::OwnedWrapper<std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> >,
std::__Cr::default_delete<std::__Cr::unique_ptr<bool, std::__Cr::default_delete<bool> > > > >, void
()>::RunOnce(base::internal::BindStateBase*) base/bind_internal.h:748:12
    #13 0x7f9c66b538ee in Run base/callback.h:142:12
    #14 0x7f9c66b538ee in base::(anonymous namespace)::PostTaskAndReplyRelay::RunReply(base::(anonymous
namespace)::PostTaskAndReplyRelay) base/threading/post_task_and_reply_impl.cc:118:29
    #15 0x7f9c66b53b4b in Invoke<void (*)(base::(anonymous namespace)::PostTaskAndReplyRelay), base::(anonymous
namespace)::PostTaskAndReplyRelay> base/bind_internal.h:437:12
    #16 0x7f9c66b53b4b in MakeItSo<void (*)(base::(anonymous namespace)::PostTaskAndReplyRelay), base::(anonymous
namespace)::PostTaskAndReplyRelay> base/bind_internal.h:706:12
    #17 0x7f9c66b53b4b in RunImpl<void (*)(base::(anonymous namespace)::PostTaskAndReplyRelay),
std::__Cr::tuple<base::(anonymous namespace)::PostTaskAndReplyRelay>, 0UL> base/bind_internal.h:779:12
    #18 0x7f9c66b53b4b in base::internal::Invoker<base::internal::BindState<void (*)(base::(anonymous
namespace)::PostTaskAndReplyRelay), base::(anonymous namespace)::PostTaskAndReplyRelay>, void
()>::RunOnce(base::internal::BindStateBase*) base/bind_internal.h:748:12
    #19 0x7f9c66a4b099 in Run base/callback.h:142:12
    #20 0x7f9c66a4b099 in base::TaskAnnotator::RunTaskImpl(base::PendingTask&)
base/task/common/task_annotator.cc:135:32
    #21 0x7f9c66acb844 in RunTask<(lambda at
../../base/task/sequence_manager/thread_controller_with_message_pump_impl.cc:388:29)>
base/task/common/task_annotator.h:74:5
    #22 0x7f9c66acb844 in
base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWorkImpl(base::sequence_manager::Lazy
Now*) base/task/sequence_manager/thread_controller_with_message_pump_impl.cc:386:21
    #23 0x7f9c66aca326 in base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork()
base/task/sequence_manager/thread_controller_with_message_pump_impl.cc:291:41

    #24 0x7f9c66acc9d1 in non-virtual thunk to
base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork()
base/task/sequence_manager/thread_controller_with_message_pump_impl.cc
```

```
base/task/sequence_manager/thread_controller_with_message_pump_impl.cc
   #25 0x7f9c66c7ab35 in base::MessagePumpLibevent::Run(base::MessagePump::Delegate*)
base/message_loop/message_pump_libevent.cc:195:55
   #26 0x7f9c66acd7fa in base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run(bool,
base::TimeDelta) base/task/sequence_manager/thread_controller_with_message_pump_impl.cc:498:12
   #27 0x7f9c66990cca in base::RunLoop::Run(base::Location const&) base/run_loop.cc:141:14
   #28 0x7f9c3ec4acb1 in content::BrowserMainLoop::RunMainMessageLoop()
content/browser/browser_main_loop.cc:1067:18
   #29 0x7f9c3ec505d7 in content::BrowserMainRunnerImpl::Run() content/browser/browser_main_runner_impl.cc:155:15
   #30 0x7f9c3ec4445a in content::BrowserMain(content::MainFunctionParams) content/browser/browser_main.cc:30:28
   #31 0x7f9c41272fe9 in content::RunBrowserProcessMain(content::MainFunctionParams, content::ContentMainDelegate*)
content/app/content_main_runner_impl.cc:641:10
   #32 0x7f9c41276472 in content::ContentMainRunnerImpl::RunBrowser(content::MainFunctionParams, bool)
content/app/content_main_runner_impl.cc:1148:10
   #33 0x7f9c41275782 in content::ContentMainRunnerImpl::Run() content/app/content_main_runner_impl.cc:1020:12
   #34 0x7f9c4126efb9 in content::RunContentProcess(content::ContentMainParams, content::ContentMainRunner*)
content/app/content_main.cc:407:36
   #35 0x7f9c4126f64f in content::ContentMain(content::ContentMainParams) content/app/content_main.cc:435:10
   #36 0x555bdb2f2fb5 in ChromeMain chrome/app/chrome_main.cc:176:12
   #37 0x7f9bf559f0b2 in __libc_start_main /build/glibc-sMfBJT/glibc-2.31/csu/../csu/libc-start.c:308:16

0x61f000221280 is located 0 bytes inside of 3112-byte region [0x61f000221280,0x61f000221ea8)
freed by thread T0 (chrome) here:
   #0 0x555bdb2f100d in operator delete(void*) /b/s/w/ir/cache/builder/src/third_party/llvm/compiler-
rt/lib/asan/asan_new_delete.cpp:152:3
   #1 0x555be7df2bb3 in operator() buildtools/third_party/libc++/trunk/include/__memory/unique_ptr.h:54:5
   #2 0x555be7df2bb3 in reset buildtools/third_party/libc++/trunk/include/__memory/unique_ptr.h:315:7
   #3 0x555be7df2bb3 in TabStripModel::SendDetachWebContentsNotifications(TabStripModel::DetachNotifications*)
chrome/browser/ui/tabs/tab_strip_model.cc:569:27
   #4 0x555be7df8c4f in TabStripModel::CloseTabs(base::span<content::WebContents* const, 18446744073709551615ul>,
unsigned int) chrome/browser/ui/tabs/tab_strip_model.cc:1919:5
   #5 0x555be7df961e in TabStripModel::CloseWebContentsAt(int, unsigned int)
chrome/browser/ui/tabs/tab_strip_model.cc:787:10
   #6 0x555be8e8b325 in BrowserTabStripController::CloseTab(int)
chrome/browser/ui/views/tabs/browser_tab_strip_controller.cc:331:11
   #7 0x555be8f2266d in TabStrip::CloseTabInternal(int, CloseTabSource)
chrome/browser/ui/views/tabs/tab_strip.cc:1784:16
   #8 0x555be8f221e6 in TabStrip::CloseTab(Tab*, CloseTabSource) chrome/browser/ui/views/tabs/tab_strip.cc:1274:3
   #9 0x555be8e9ea23 in Tab::CloseButtonPressed(ui::Event const&) chrome/browser/ui/views/tabs/tab.cc:1090:16
   #10 0x7f9c502c158e in views::Button::DefaultButtonControllerDelegate::NotifyClick(ui::Event const&)
ui/views/controls/button/button.cc:67:13
   #11 0x7f9c502ca8ee in views::ButtonController::OnMouseReleased(ui::MouseEvent const&)
ui/views/controls/button/button_controller.cc
   #12 0x7f9c513128af in ui::ScopedTargetHandler::OnEvent(ui::Event*) ui/events/scoped_target_handler.cc:28:24
   #13 0x7f9c51301941 in ui::EventDispatcher::DispatchEvent(ui::EventHandler*, ui::Event*)
ui/events/event_dispatcher.cc:190:12
   #14 0x7f9c51300b02 in ui::EventDispatcher::ProcessEvent(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:139:5
   #15 0x7f9c513005d6 in ui::EventDispatcherDelegate::DispatchEventToTarget(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:83:14
   #16 0x7f9c5130033e in ui::EventDispatcherDelegate::DispatchEvent(ui::EventTarget*, ui::Event*)

ui/events/event_dispatcher.cc:55:15
   #17 0x7f9c5050a98a in views::internal::RootView::OnMouseReleased(ui::MouseEvent const&)
ui/views/widget/root_view.cc:485:9
```

```
ui/views/widget/root_view.cc:485:9
    #18 0x7f9c50521492 in views::Widget::OnMouseEvent(ui::MouseEvent*) ui/views/widget/widget.cc:1566:20
    #19 0x7f9c5058c04d in views::NativeWidgetAura::OnMouseEvent(ui::MouseEvent*)
ui/views/widget/native_widget_aura.cc
    #20 0x7f9c51301941 in ui::EventDispatcher::DispatchEvent(ui::EventHandler*, ui::Event*)
ui/events/event_dispatcher.cc:190:12
    #21 0x7f9c51300b02 in ui::EventDispatcher::ProcessEvent(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:139:5
    #22 0x7f9c513005d6 in ui::EventDispatcherDelegate::DispatchEventToTarget(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:83:14
    #23 0x7f9c5130033e in ui::EventDispatcherDelegate::DispatchEvent(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:55:15
    #24 0x7f9c51304ce9 in ui::EventProcessor::OnEventFromSource(ui::Event*) ui/events/event_processor.cc:49:17
    #25 0x7f9c51307b7c in ui::EventSource::DeliverEventToSink(ui::Event*) ui/events/event_source.cc:118:16
    #26 0x7f9c5130807c in ui::EventSource::EventRewriterContinuationImpl::SendEvent(ui::Event const*)
ui/events/event_source.cc:66:14
    #27 0x7f9c51306345 in ui::EventRewriter::SendEvent(base::WeakPtr<ui::EventRewriterContinuation>, ui::Event const*)
ui/events/event_rewriter.cc:88:39
    #28 0x7f9c4d985a2d in ui::EventRewriterChromeOS::RewriteMouseButtonEvent(ui::MouseEvent const&,
base::WeakPtr<ui::EventRewriterContinuation>) ui/chromeos/events/event_rewriter_chromeos.cc:1273:12
    #29 0x7f9c4d985f6d in ui::EventRewriterChromeOS::RewriteEvent(ui::Event const&,
base::WeakPtr<ui::EventRewriterContinuation>) ui/chromeos/events/event_rewriter_chromeos.cc:757:12
    #30 0x7f9c5130802c in ui::EventSource::EventRewriterContinuationImpl::SendEvent(ui::Event const*)
ui/events/event_source.cc:67:32
    #31 0x7f9c51306345 in ui::EventRewriter::SendEvent(base::WeakPtr<ui::EventRewriterContinuation>, ui::Event const*)
ui/events/event_rewriter.cc:88:39

previously allocated by thread T0 (chrome) here:
    #0 0x555bdb2f07ad in operator new(unsigned long) /b/s/w/ir/cache/builder/src/third_party/llvm/compiler-
rt/lib/asan/asan_new_delete.cpp:95:3
    #1 0x7f9c40003e81 in content::WebContents::CreateWithSessionStorage(content::WebContents::CreateParams const&,
std::__Cr::map<content::StoragePartitionConfig, scoped_refptr<content::SessionStorageNamespace>,
std::__Cr::less<content::StoragePartitionConfig>, std::__Cr::allocator<std::__Cr::pair<content::StoragePartitionConfig const,
scoped_refptr<content::SessionStorageNamespace> > > > const&)
content/browser/web_contents/web_contents_impl.cc:601:7
    #2 0x555be7d042ae in chrome::(anonymous namespace)::CreateRestoredTab(Browser*,
std::__Cr::vector<sessions::SerializedNavigationEntry, std::__Cr::allocator<sessions::SerializedNavigationEntry> > const&,
int, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > const&, base::TimeTicks,
content::SessionStorageNamespace*, sessions::SerializedUserAgentOverride const&,
std::__Cr::map<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >,
std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >,
std::__Cr::less<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > >,
std::__Cr::allocator<std::__Cr::pair<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >
const, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > > > > const&, bool, bool)
chrome/browser/ui/browser_tabrestore.cc:80:7
    #3 0x555be7d03dcd in chrome::AddRestoredTab(Browser*, std::__Cr::vector<sessions::SerializedNavigationEntry,
std::__Cr::allocator<sessions::SerializedNavigationEntry> > const&, int, int, std::__Cr::basic_string<char,
std::__Cr::char_traits<char>, std::__Cr::allocator<char> > const&, absl::optional<tab_groups::TabGroupId>, bool, bool,
base::TimeTicks, content::SessionStorageNamespace*, sessions::SerializedUserAgentOverride const&,
std::__Cr::map<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >,
std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >,

std::__Cr::less<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > >,
std::__Cr::allocator<std::__Cr::pair<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >
const, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > > > > const&, bool)
```

```
const, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > > > > const&, bool)
chrome/browser/ui/browser_tabrestore.cc:238:47
   #4 0x555be7cf4396 in BrowserLiveTabContext::AddRestoredTab(std::__Cr::vector<sessions::SerializedNavigationEntry,
std::__Cr::allocator<sessions::SerializedNavigationEntry> > const&, int, int, std::__Cr::basic_string<char,
std::__Cr::char_traits<char>, std::__Cr::allocator<char> > const&, absl::optional<tab_groups::TabGroupId>,
tab_groups::TabGroupVisualData const&, bool, bool, sessions::PlatformSpecificTabData const*,
sessions::SerializedUserAgentOverride const&, std::__Cr::map<std::__Cr::basic_string<char, std::__Cr::char_traits<char>,
std::__Cr::allocator<char> >, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >,
std::__Cr::less<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > >,
std::__Cr::allocator<std::__Cr::pair<std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> >
const, std::__Cr::basic_string<char, std::__Cr::char_traits<char>, std::__Cr::allocator<char> > > > > const&, SessionID
const*) chrome/browser/ui/browser_live_tab_context.cc:218:20
   #5 0x7f9c31ddf1a3 in sessions::TabRestoreServiceHelper::RestoreTab(sessions::TabRestoreService::Tab const&,
sessions::LiveTabContext*, WindowOpenDisposition, sessions::LiveTab**)
components/sessions/core/tab_restore_service_helper.cc:894:29
   #6 0x7f9c31ddb5a9 in sessions::TabRestoreServiceHelper::RestoreEntryById(sessions::LiveTabContext*, SessionID,
WindowOpenDisposition) components/sessions/core/tab_restore_service_helper.cc:477:17
   #7 0x7f9c31df497d in sessions::TabRestoreServiceImpl::RestoreEntryById(sessions::LiveTabContext*, SessionID,
WindowOpenDisposition) components/sessions/core/tab_restore_service_impl.cc:1498:18
   #8 0x555be8ab97c2 in RecentTabsSubMenuModel::ExecuteCommand(int, int)
chrome/browser/ui/toolbar/recent_tabs_sub_menu_model.cc:255:18
   #9 0x555be8f64f44 in AppMenu::ExecuteCommand(int, int) chrome/browser/ui/views/toolbar/app_menu.cc:1011:23
   #10 0x7f9c5035c8aa in
views::internal::MenuRunnerImpl::OnMenuClosed(views::internal::MenuControllerDelegate::NotifyType,
views::MenuItemView*, int) ui/views/controls/menu/menu_runner_impl.cc:233:29
   #11 0x7f9c5032212e in views::MenuController::ExitMenu() ui/views/controls/menu/menu_controller.cc:3179:13
   #12 0x7f9c5032fb5b in views::MenuController::ReallyAccept(views::MenuItemView*, int)
ui/views/controls/menu/menu_controller.cc:1791:3
   #13 0x7f9c503269e2 in views::MenuController::OnMouseReleased(views::SubmenuView*, ui::MouseEvent const&)
ui/views/controls/menu/menu_controller.cc:828:7
   #14 0x7f9c50521492 in views::Widget::OnMouseEvent(ui::MouseEvent*) ui/views/widget/widget.cc:1566:20
   #15 0x7f9c5058c04d in views::NativeWidgetAura::OnMouseEvent(ui::MouseEvent*)
ui/views/widget/native_widget_aura.cc
   #16 0x7f9c51301941 in ui::EventDispatcher::DispatchEvent(ui::EventHandler*, ui::Event*)
ui/events/event_dispatcher.cc:190:12
   #17 0x7f9c51300b02 in ui::EventDispatcher::ProcessEvent(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:139:5
   #18 0x7f9c513005d6 in ui::EventDispatcherDelegate::DispatchEventToTarget(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:83:14
   #19 0x7f9c5130033e in ui::EventDispatcherDelegate::DispatchEvent(ui::EventTarget*, ui::Event*)
ui/events/event_dispatcher.cc:55:15
   #20 0x7f9c51304ce9 in ui::EventProcessor::OnEventFromSource(ui::Event*) ui/events/event_processor.cc:49:17
   #21 0x7f9c51307b7c in ui::EventSource::DeliverEventToSink(ui::Event*) ui/events/event_source.cc:118:16
   #22 0x7f9c5130807c in ui::EventSource::EventRewriterContinuationImpl::SendEvent(ui::Event const*)
ui/events/event_source.cc:66:14
   #23 0x7f9c51306345 in ui::EventRewriter::SendEvent(base::WeakPtr<ui::EventRewriterContinuation>, ui::Event const*)
ui/events/event_rewriter.cc:88:39
   #24 0x7f9c4d985a2d in ui::EventRewriterChromeOS::RewriteMouseButtonEvent(ui::MouseEvent const&,
base::WeakPtr<ui::EventRewriterContinuation>) ui/chromeos/events/event_rewriter_chromeos.cc:1273:12
   #25 0x7f9c4d985f6d in ui::EventRewriterChromeOS::RewriteEvent(ui::Event const&,
base::WeakPtr<ui::EventRewriterContinuation>) ui/chromeos/events/event_rewriter_chromeos.cc:757:12

   #26 0x7f9c5130802c in ui::EventSource::EventRewriterContinuationImpl::SendEvent(ui::Event const*)
ui/events/event_source.cc:67:32
   #27 0x7f9c51306345 in ui::EventRewriter::SendEvent(base::WeakPtr<ui::EventRewriterContinuation>, ui::Event const*)
```

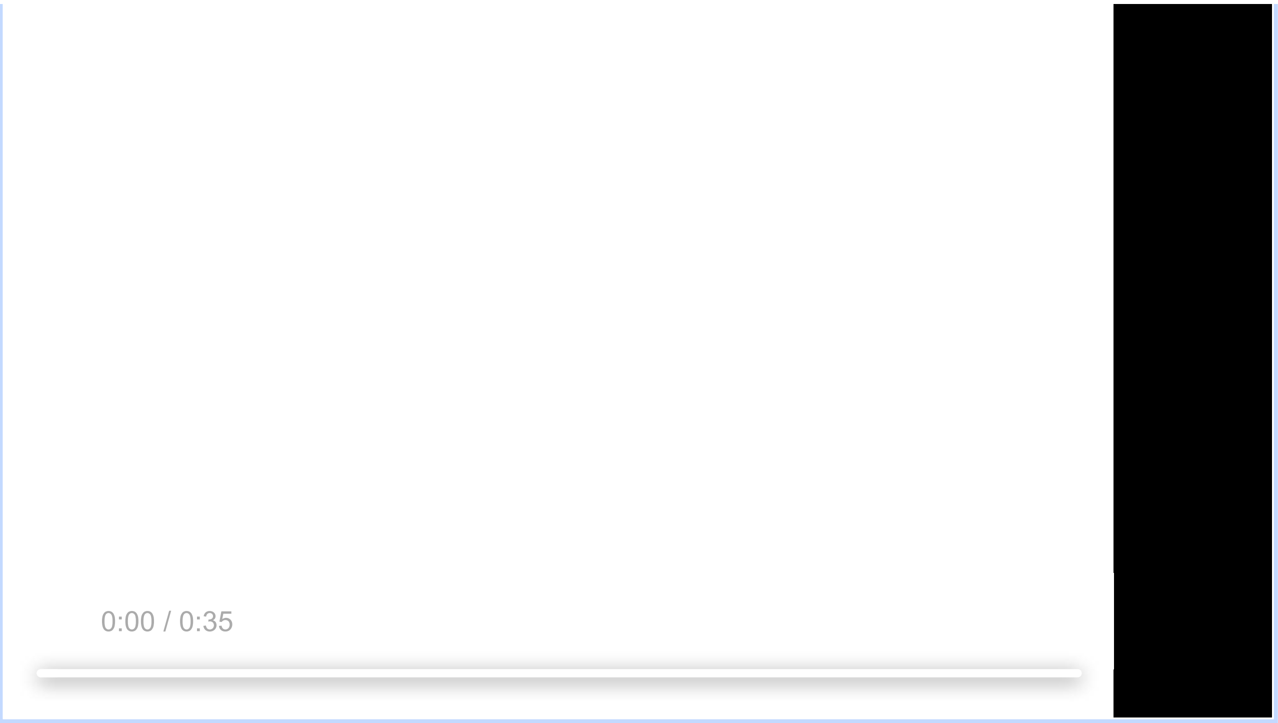
```
    #27 0x7f9c51306345 in ui::EventRewriter::SendEvent(base::WeakPtr<ui::EventRewriterContinuation>, ui::Event const*)
ui/events/event_rewriter.cc:88:39
    #28 0x7f9c4c7a7db2 in ash::KeyboardDrivenEventRewriter::RewriteEvent(ui::Event const&,
base::WeakPtr<ui::EventRewriterContinuation>) ash/events/keyboard_driven_event_rewriter.cc:31:12
    #29 0x7f9c5130802c in ui::EventSource::EventRewriterContinuationImpl::SendEvent(ui::Event const*)
ui/events/event_source.cc:67:32

SUMMARY: AddressSanitizer: heap-use-after-free chrome/browser/ash/crostini/crostini_export_import.cc:200:21 in
crostini::CrostiniExportImport::OpenFileDialog(crostini::CrostiniExportImport::OperationData*, content::WebContents*)
Shadow bytes around the buggy address:
  0x0c3e8003c200: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c3e8003c210: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c3e8003c220: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c3e8003c230: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c3e8003c240: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c3e8003c250:[fd]fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c3e8003c260: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c3e8003c270: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c3e8003c280: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c3e8003c290: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c3e8003c2a0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
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
==519913==ABORTING
```

Did this work before? N/A

Chrome version: 91.0.4472.101 Channel: n/a
OS Version:

uaf1.mp4
12.0 MB View Download

[redacted]



asan.log
21.5 KB View Download

Comment 1 by sheriffbot on Mon, Mar 28, 2022, 5:22 AM EDT

Labels: external_security_report

Comment 2 by etern...@gmail.com on Mon, Mar 28, 2022, 5:26 AM EDT

other:
Because CrostiniUpgrader is KeyedService, It will only be free after ChromeOS is closed.
So even if the webcontent is free, the OnBackupPathChecked callback will still happen.

Comment 3 by etern...@gmail.com on Mon, Mar 28, 2022, 5:28 AM EDT

CrostiniUpgrader::Restore has the same problem.

Comment 4 Deleted

Comment 5 by etern...@gmail.com on Mon, Mar 28, 2022, 5:35 AM EDT

Since backup can be called directly through the mojo interface [0], a compromised renderer may trigger the UAF without interaction(maybe?).

```
interface PageHandler {
  // Backup the existing container. If |show_file_chooser| is true, the
  // user will be shown a dialog to decide where to store the backup.
  Backup(bool show_file_chooser);
  // Start running upgrade prechecks. Result is asynchronously
  // returned via Page::PrecheckStatus.

```

[0]

[0]
https://source.chromium.org/chromium/chromium/src/+/main:chrome/browser/ui/webui/chromeos/crostini_upgrader/crostini_upgrader.mojom;l=30;drc=8a6efc8a06137639ea50513b0ba62932e9698038

Comment 6 by anunoy@chromium.org on Mon, Mar 28, 2022, 8:40 AM EDT

Labels: -OS-Linux OS-Chrome

Comment 7 by aashay@google.com on Mon, Mar 28, 2022, 6:14 PM EDT

Status: Assigned (was: Unconfirmed)
Owner: sidereal@google.com
Cc: aashay@chromium.org
Labels: Security_Impact-Stable Security_Severity-High FoundIn-91 Pri-1
Components: UI>Shell

Severity-high because this is a UAF in the browser process but mitigated by requiring user action (clicking upgrade).

(Assigning an owner based on file modification history)

Comment 8 by sheriffbot on Mon, Mar 28, 2022, 6:16 PM EDT

Labels: -Security_Impact-Stable Security_Impact-Extended

Comment 9 by sheriffbot on Tue, Mar 29, 2022, 12:46 PM EDT

Labels: M-98 Target-98

Setting milestone and target because of high severity.

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

Comment 10 by etern...@gmail.com on Wed, Mar 30, 2022, 5:43 AM EDT

sorry, I'm not sure when this vulnerability was introduced.

I build chromeos in commit:
```
commit 6a9e83b727d89b4937b25764c7a793b677fc2480 (HEAD -> main, origin/main, origin/lkgr-android-internal, origin/HEAD)
Author: Chrome Release Bot (LUCI) <chrome-official-brancher@chops-service-accounts.iam.gserviceaccount.com>
Date:   Sun Mar 27 03:01:56 2022 +0000
```

build args:
```
is_asan = true
is_debug = false
enable_nacl = false
is_component_build = true
symbol_level=1
target_os="chromeos"
```

Comment 11 by sheriffbot on Wed, Mar 30, 2022, 12:21 PM EDT

Labels: -M-98 M-100 Target-100

Comment 12 by etern...@gmail.com on Thu, Mar 31, 2022, 9:57 PM EDT

Is anyone working on this issue? thanks :)

Comment 13 by sidereal@google.com on Thu, Mar 31, 2022, 10:29 PM EDT

Probably the fix here is to use content::WebContents::GetWeakPtr() to get a weak pointer, and just abandon the operation if it becomes null.

Comment 14 by Git Watcher on Fri, Apr 1, 2022, 2:33 AM EDT

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/19d06d3fb2d9824e06f3dcac492815d785910fd9

commit 19d06d3fb2d9824e06f3dcac492815d785910fd9
Author: Fergus Dall <sidereal@google.com>
Date: Fri Apr 01 06:32:55 2022

crostini_upgrader: Handle content::WebContents through weak pointers

content::WebContents is not owned by the upgrader but gets passed to
callbacks, so we need to be able to check if it has been destroyed.

~~Bug: 1310717~~
Change-Id: I343d0361c39d190c070e1a693cc5d692d0121071
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565219
Reviewed-by: Timothy Loh <timloh@chromium.org>
Commit-Queue: Fergus Dall <sidereal@google.com>
Cr-Commit-Position: refs/heads/main@{#987790}

[modify] https://crrev.com/19d06d3fb2d9824e06f3dcac492815d785910fd9/chrome/browser/ash/crostini/crostini_upgrader_ui_delegate.h
[modify] https://crrev.com/19d06d3fb2d9824e06f3dcac492815d785910fd9/chrome/browser/ui/webui/chromeos/crostini_upgrader/crostini_upgrader_page_handler.cc
[modify] https://crrev.com/19d06d3fb2d9824e06f3dcac492815d785910fd9/chrome/browser/ash/crostini/crostini_upgrader.h
[modify] https://crrev.com/19d06d3fb2d9824e06f3dcac492815d785910fd9/chrome/browser/ash/crostini/crostini_upgrader.cc

Comment 15 by sidereal@google.com on Fri, Apr 1, 2022, 9:48 AM EDT

Labels: -Target-98 Merge-Request-101 Merge-Request-100

Requesting merge to stable per ChromeOS Security SLOs

Comment 16 by aashay@google.com on Fri, Apr 1, 2022, 4:27 PM EDT

Labels: -Merge-Request-101

Comment 17 by sheriffbot on Sat, Apr 2, 2022, 2:37 AM EDT

Labels: -Merge-Request-100 Hotlist-Merge-Review Merge-Review-100

Merge review required: M100 is already shipping to stable.

Please answer the following questions so that we can safely process your merge request:

Please answer the following questions so that we can safely process your merge request:
1. Why does your merge fit within the merge criteria for these milestones?
- Chrome Browser: https://chromiumdash.appspot.com/branches
- Chrome OS: https://goto.google.com/cros-release-branch-merge-guidelines
2. What changes specifically would you like to merge? Please link to Gerrit.
3. Have the changes been released and tested on canary?
4. Is this a new feature? If yes, is it behind a Finch flag and are experiments active in any release channels?
5. [Chrome OS only]: Was the change reviewed and approved by the Eng Prod Representative?
https://goto.google.com/cros-engprodcomponents
6. If this merge addresses a major issue in the stable channel, does it require manual verification by the test team? If so, please describe required testing.

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

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

Comment 18 by sidereal@google.com on Sat, Apr 2, 2022, 4:30 AM EDT

1) Yes - High severity security bugs get merged to stable
2) https://chromium-review.googlesource.com/c/chromium/src/+/3565219
3) No, this is waiting on a chrome uprev
4) No
5) N/A
6) No

Comment 19 by etern...@gmail.com on Sat, Apr 2, 2022, 10:49 PM EDT

Can I get a cve and bug bounty?
credit: Nan Wang(@eternalsakura13) and Guang Gong of 360 Alpha Lab

Comment 20 by dgagnon@google.com on Mon, Apr 4, 2022, 3:46 PM EDT

RE comment #18, please update once the fix is confirmed on a canary/ToT build. I'll review again after that for merge to M100.

Comment 21 by aashay@google.com on Mon, Apr 4, 2022, 6:04 PM EDT

Re #c19: This issue will be routed to our VRP panel once the fix lands and this issue is marked as fixed.

Comment 22 by sidereal@google.com on Thu, Apr 7, 2022, 12:57 PM EDT

Cc: dgagnon@chromium.org

RE comment 20: Tested with Chrome OS build 14673.0.0, fix is effective.

Comment 23 by dgagnon@google.com on Thu, Apr 7, 2022, 12:59 PM EDT

Labels: -Hotlist-Merge-Review -Merge-Review-100 Merge-Approved-100

Approved for M100

Comment 24 by Git Watcher on Thu, Apr 7, 2022, 11:10 PM EDT

Labels: -merge-approved-100 merge-merged-4896 merge-merged-100

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed

commit 29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed
Author: Fergus Dall <sidereal@google.com>
Date: Fri Apr 08 03:09:35 2022

crostini_upgrader: Handle content::WebContents through weak pointers

content::WebContents is not owned by the upgrader but gets passed to
callbacks, so we need to be able to check if it has been destroyed.

(cherry picked from commit 19d06d3fb2d9824e06f3dcac492815d785910fd9)

~~Bug: 1310717~~
Change-Id: I343d0361c39d190c070e1a693cc5d692d0121071
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565219
Reviewed-by: Timothy Loh <timloh@chromium.org>
Commit-Queue: Fergus Dall <sidereal@google.com>
Cr-Original-Commit-Position: refs/heads/main@{#987790}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565520
Auto-Submit: Fergus Dall <sidereal@google.com>
Reviewed-by: David Munro <davidmunro@google.com>
Commit-Queue: David Munro <davidmunro@google.com>
Cr-Commit-Position: refs/branch-heads/4896@{#1079}
Cr-Branched-From: 1f63ff4bc27570761b35ffbc7f938f6586f7bee8-refs/heads/main@{#972766}

[modify] https://crrev.com/29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed/chrome/browser/ash/crostini/crostini_upgrader_ui_delegate.h
[modify] https://crrev.com/29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed/chrome/browser/ash/crostini/crostini_upgrader.h
[modify] https://crrev.com/29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed/chrome/browser/ui/webui/chromeos/crostini_upgrader/crostini_upgrader_page_handler.cc
[modify] https://crrev.com/29a0add3cb1bf14cc3fadb7c9c55b9f3a0fc5eed/chrome/browser/ash/crostini/crostini_upgrader.cc

Comment 25 by Git Watcher on Thu, Apr 7, 2022, 11:29 PM EDT

Labels: merge-merged-4951 merge-merged-101

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef

commit 18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef
Author: Fergus Dall <sidereal@google.com>
Date: Fri Apr 08 03:28:17 2022

crostini_upgrader: Handle content::WebContents through weak pointers

content::WebContents is not owned by the upgrader but gets passed to
callbacks, so we need to be able to check if it has been destroyed.

(cherry picked from commit 19d06d3fb2d9824e06f3dcac492815d785910fd9)

~~Bug: 1310717~~
Change-Id: I343d0361c39d190c070e1a693cc5d692d0121071
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565219

Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565219
Reviewed-by: Timothy Loh <timloh@chromium.org>
Commit-Queue: Fergus Dall <sidereal@google.com>
Cr-Original-Commit-Position: refs/heads/main@{#987790}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565583
Auto-Submit: Fergus Dall <sidereal@google.com>
Reviewed-by: David Munro <davidmunro@google.com>
Commit-Queue: David Munro <davidmunro@google.com>
Cr-Commit-Position: refs/branch-heads/4951@{#566}
Cr-Branched-From: 27de6227ca357da0d57ae2c7b18da170c4651438-refs/heads/main@{#982481}

[modify] https://crrev.com/18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef/chrome/browser/ash/crostini/crostini_upgrader_ui_delegate.h
[modify] https://crrev.com/18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef/chrome/browser/ui/webui/chromeos/crostini_upgrader/crostini_upgrader_page_handler.cc
[modify] https://crrev.com/18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef/chrome/browser/ash/crostini/crostini_upgrader.h
[modify] https://crrev.com/18c12392ffd2cd2a88e7f75d62c5f9364a4d0cef/chrome/browser/ash/crostini/crostini_upgrader.cc

Comment 26 by sidereal@google.com on Thu, Apr 7, 2022, 11:30 PM EDT

Status: Fixed (was: Assigned)

Comment 27 by sheriffbot on Thu, Apr 7, 2022, 11:32 PM EDT

Labels: LTS-Merge-Candidate

LTS Milestone M96

This issue has been flagged as a merge candidate for Chrome OS' LTS channel. If selected, our merge team will handle any additional merges. To help us determine if this issue requires a merge to LTS, please answer this short questionnaire:
1. Was this issue a regression for the milestone it was found in?
2. Is this issue related to a change or feature merged after the latest LTS Milestone?

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

Comment 28 by sidereal@google.com on Thu, Apr 7, 2022, 11:36 PM EDT

This issue will be present in the same form in M96, and was introduced well before then

Comment 29 by sheriffbot on Sat, Apr 9, 2022, 12:41 PM EDT

Labels: reward-topanel

Comment 30 by sheriffbot on Sat, Apr 9, 2022, 1:40 PM EDT

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

Comment 31 by rzanoni@google.com on Mon, Apr 11, 2022, 12:20 PM EDT

Cc: rzanoni@google.com
Labels: LTS-Evaluating-96

Comment 32 by adetaylor@google.com on Mon, Apr 11, 2022, 1:15 PM EDT

Labels: Release-2-M100

Comment 33 by adetaylor@google.com on Mon, Apr 11, 2022, 1:30 PM EDT

Labels: CVE-2022-1311 CVE_description-missing

Comment 34 by rzanoni@google.com on Tue, Apr 12, 2022, 8:26 AM EDT

Labels: -LTS-Evaluating-96 LTS-Merge-Request-96

Comment 35 by sheriffbot on Tue, Apr 12, 2022, 8:27 AM EDT

Labels: -LTS-Merge-Request-96 LTS-Merge-Review-96

This issue requires additional review before it can be merged to the LTS channel. Please answer the following questions to help us evaluate this merge:

1. Number of CLs needed for this fix and links to them.
2. Level of complexity (High, Medium, Low - Explain)
3. Has this been merged to a stable release? beta release?
4. Overall Recommendation (Yes, No)

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

Comment 36 by rzanoni@google.com on Tue, Apr 12, 2022, 8:37 AM EDT

1. Just https://crrev.com/c/3581703
2. Low, no conflicts
3. 100, 101
4. Yes

Comment 37 by gmpritchard@google.com on Wed, Apr 13, 2022, 9:15 AM EDT

Labels: -LTS-Merge-Candidate -LTS-Merge-Review-96 LTS-Merge-Approved-96

Comment 38 by Git Watcher on Wed, Apr 13, 2022, 12:38 PM EDT

Labels: merge-merged-4664

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/5dd954d87a0c091a7874b40d39f27375fcea10fb

commit 5dd954d87a0c091a7874b40d39f27375fcea10fb
Author: Fergus Dall <sidereal@google.com>
Date: Wed Apr 13 16:37:30 2022

[M96-LTS] crostini_upgrader: Handle content::WebContents through weak pointers

content::WebContents is not owned by the upgrader but gets passed to callbacks, so we need to be able to check if it has been destroyed.

(cherry picked from commit 19d06d3fb2d9824e06f3dcac492815d785910fd9)

~~Bug: 1310717~~
Change-Id: I343d0361c39d190c070e1a693cc5d692d0121071
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3565219
Commit-Queue: Fergus Dall <sidereal@google.com>
Cr-Original-Commit-Position: refs/heads/main@{#987790}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3581703
Reviewed-by: Artem Sumaneev <asumaneev@google.com>
Owners-Override: Artem Sumaneev <asumaneev@google.com>
Commit-Queue: Roger Felipe Zanoni da Silva <rzanoni@google.com>
Cr-Commit-Position: refs/branch-heads/4664@{#1585}
Cr-Branched-From: 24dc4ee75e01a29d390d43c9c264372a169273a7-refs/heads/main@{#929512}

[modify]
https://crrev.com/5dd954d87a0c091a7874b40d39f27375fcea10fb/chrome/browser/ash/crostini/crostini_upgrader_ui_delegate.h
[modify] https://crrev.com/5dd954d87a0c091a7874b40d39f27375fcea10fb/chrome/browser/ash/crostini/crostini_upgrader.h
[modify]
https://crrev.com/5dd954d87a0c091a7874b40d39f27375fcea10fb/chrome/browser/ui/webui/chromeos/crostini_upgrader/crostini_upgrader_page_handler.cc
[modify]
https://crrev.com/5dd954d87a0c091a7874b40d39f27375fcea10fb/chrome/browser/ash/crostini/crostini_upgrader.cc

Comment 39 by rzanoni@google.com on Wed, Apr 13, 2022, 1:03 PM EDT

Labels: -LTS-Merge-Approved-96 LTS-Merge-Merged-96

Comment 40 by amyressler@google.com on Thu, Apr 21, 2022, 8:40 PM EDT

Labels: -reward-topanel reward-unpaid reward-7000

*** 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 41 by amyressler@chromium.org on Thu, Apr 21, 2022, 9:28 PM EDT

Congratulations, Nan Wang and Guang Gong! The VRP Panel has decided to award you $7,000 for this very detailed report. While this issue does manifest in browser process memory corruption, it is mitigated by some direct and specific user interaction. If you can demonstrate exploitation through the scenario described in comment #5, we would be happy to reassess for a potential change in reward amount. Thank you for your efforts and nice work!

Comment 42 by amyressler@google.com on Mon, Apr 25, 2022, 4:17 PM EDT

Labels: -reward-unpaid reward-inprocess

Comment 43 by sheriffbot on Fri, Jul 15, 2022, 1:31 PM EDT

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

Comment 44 by amyressler@google.com on Tue, Jul 26, 2022, 4:57 PM EDT

Labels: CVE_description-submitted -CVE_description-missing

Comment 45 by amyressler@chromium.org on Fri, Jul 29, 2022, 5:26 PM EDT

Labels: -CVE_description-missing --CVE_description-missing