Starred by 1 user

Owner:	nazab...@microsoft.com
CC:	vasilyt@chromium.org sande...@chromium.org pbomm...@chromium.org adetaylor@google.com
Status:	Fixed *(Closed)*
Components:	Internals>Media
Modified:	Jan 5, 2022
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux
Pri:	1
Type:	Bug-Security

reward-5000
Security_Impact-Stable
Arch-x86_64
Hotlist-Merge-Approved
Security_Severity-High
allpublic
reward-inprocess
Unreproducible
Via-Wizard-Security
CVE_description-submitted
Target-88
Target-87
M-88
merge-merged-4240
merge-merged-86
LTR-Merged-86
LTS-Security-86
merge-merged-4324
merge-merged-88
external_security_report
merge-merged-4389
merge-merged-89
Release-3-M88
CVE-2021-21152

Issue 1166504: heap bufferoverflow in VideoFrameYUVConverter

Reported by emily...@gmail.com on Thu, Jan 14, 2021, 1:06 AM EST

Code

UserAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36

Steps to reproduce the problem:
Ubuntu 20.04
Chromium 89.0.4381.6
Version 89.0.4388.0 (Developer Build) (64-bit)
./chrome --enable-experimental-web-platform-features http://localhost:8000/crash.html

What is the expected behavior?

What went wrong?
==1==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62b000053b78 at pc 0x55e94c339bf7 bp 0x7fff6ff74d20 sp 0x7fff6ff744e8
READ of size 27358 at 0x62b000053b78 thread T0 (chrome)
error: unknown argument '--demangle=True'
    #0 0x55e94c339bf6 in __asan_memcpy /b/s/w/ir/cache/builder/src/third_party/llvm/compiler-rt/lib/asan/asan_interceptors_memintrinsics.cpp:22
    #1 0x55e94c339bf6 in ?? ??:0
    #2 0x55e95c38ac90 in gpu::gles2::GLES2Implementation::TexSubImage2DImpl(unsigned int, int, int, int, int, int, unsigned int, unsigned int, unsigned int, void const*, unsigned int, unsigned char, gpu::ScopedTransferBufferPtr*, unsigned int) ./../../gpu/command_buffer/client/gles2_implementation.cc:131
    #3 0x55e95c38ac90 in TexSubImage2DImpl ./../../gpu/command_buffer/client/gles2_implementation.cc:3918
    #4 0x55e95c38ac90 in ?? ??:0
    #5 0x55e95c38fc82 in gpu::gles2::GLES2Implementation::TexSubImage2D(unsigned int, int, int, int, int, int, unsigned int, unsigned int, void const*) ./../../gpu/command_buffer/client/gles2_implementation.cc:3720
    #6 0x55e95c38fc82 in ?? ??:0
    #7 0x55e94e83e42c in gpu::raster::RasterImplementationGLES::WritePixels(gpu::Mailbox const&, int, int, unsigned int, unsigned int, SkImageInfo const&, void const*) ./../../gpu/command_buffer/client/raster_implementation_gles.cc:183
    #8 0x55e94e83e42c in ?? ??:0
    #9 0x55e94e802482 in media::VideoFrameYUVConverter::VideoFrameYUVMailboxesHolder::VideoFrameToMailboxes(media::VideoFrame const*, viz::RasterContextProvider*, gpu::Mailbox*) ./../../media/renderers/video_frame_yuv_converter.cc:302
    #10 0x55e94e802482 in ?? ??:0
    #11 0x55e94e802a57 in media::VideoFrameYUVConverter::VideoFrameYUVMailboxesHolder::VideoFrameToSkiaTextures(media::VideoFrame const*, viz::RasterContextProvider*) ./../../media/renderers/video_frame_yuv_converter.cc:315
    #12 0x55e94e802a57 in ?? ??:0
    #13 0x55e94e80457c in media::VideoFrameYUVConverter::ConvertFromVideoFrameYUVSkia(media::VideoFrame const*, viz::RasterContextProvider*, unsigned int, unsigned int, unsigned int, unsigned int, bool, bool) ./../../media/renderers/video_frame_yuv_converter.cc:497
    #14 0x55e94e80457c in ?? ??:0
    #15 0x55e94e8041ea in media::VideoFrameYUVConverter::ConvertFromVideoFrameYUVWithGrContext(media::VideoFrame const*, viz::RasterContextProvider*, gpu::MailboxHolder const&, unsigned int, unsigned int, bool, bool) ./../../media/renderers/video_frame_yuv_converter.cc:469
    #16 0x55e94e8041ea in ?? ??:0
    #17 0x55e94e803d37 in media::VideoFrameYUVConverter::ConvertYUVVideoFrame(media::VideoFrame const*, viz::RasterContextProvider*, gpu::MailboxHolder const&, unsigned int, unsigned int, bool, bool) ./../../media/renderers/video_frame_yuv_converter.cc:413
    #18 0x55e94e803d37 in ?? ??:0
    #19 0x55e94e80398a in media::VideoFrameYUVConverter::ConvertYUVVideoFrameNoCaching(media::VideoFrame const*, viz::RasterContextProvider*, gpu::MailboxHolder const&) ./../../media/renderers/video_frame_yuv_converter.cc:390
    #20 0x55e94e80398a in ?? ??:0

    #21 0x55e96aa2563c in blink::VideoFrame::CreateImageBitmap(blink::ScriptState*, base::Optional<blink::IntRect>, blink::ImageBitmapOptions const*,
blink::ExceptionState&) ./../../third_party/blink/renderer/modules/webcodecs/video_frame.cc:579
    #22 0x55e96aa2563c in ?? ??:0
    #23 0x55e96aa26909 in non-virtual thunk to blink::VideoFrame::CreateImageBitmap(blink::ScriptState*, base::Optional<blink::IntRect>, blink::ImageBitmapOptions const*,
blink::ExceptionState&) ./../../third_party/blink/renderer/modules/webcodecs/video_frame.cc:?
    #24 0x55e96aa26909 in ?? ??:0
    #25 0x55e96aa333e0 in blink::ImageBitmapFactories::CreateImageBitmap(blink::ScriptState*, blink::ImageBitmapSource*, base::Optional<blink::IntRect>,
blink::ImageBitmapOptions const*, blink::ExceptionState&) ./../../third_party/blink/renderer/core/imagebitmap/image_bitmap_factories.cc:204
    #26 0x55e96aa333e0 in ?? ??:0
    #27 0x55e96aa22ac6 in blink::VideoFrame::createImageBitmap(blink::ScriptState*, blink::ImageBitmapOptions const*, blink::ExceptionState&)
./../../third_party/blink/renderer/modules/webcodecs/video_frame.cc:418
    #28 0x55e96aa22ac6 in ?? ??:0
    #29 0x55e96aa2d9b4 in blink::(anonymous namespace)::CreateImageBitmapOperationCallback(v8::FunctionCallbackInfo<v8::Value> const&)
./gen/third_party/blink/renderer/bindings/modules/v8/v8_video_frame.cc:340
    #30 0x55e96aa2d9b4 in ?? ??:0
    #31 0x55e954550331 in v8::internal::FunctionCallbackArguments::Call(v8::internal::CallHandlerInfo) ./../../v8/src/api/api-arguments-inl.h:158
    #32 0x55e954550331 in ?? ??:0
    #33 0x55e95454deaf in v8::internal::MaybeHandle<v8::internal::Object> v8::internal::(anonymous namespace)::HandleApiCallHelper<false>(v8::internal::Isolate*,
v8::internal::Handle<v8::internal::HeapObject>, v8::internal::Handle<v8::internal::HeapObject>, v8::internal::Handle<v8::internal::FunctionTemplateInfo>,
v8::internal::Handle<v8::internal::Object>, v8::internal::BuiltinArguments) ./../../v8/src/builtins/builtins-api.cc:113
    #34 0x55e95454deaf in ?? ??:0
    #35 0x55e95454baa8 in v8::internal::Builtin_Impl_HandleApiCall(v8::internal::BuiltinArguments, v8::internal::Isolate*) ./../../v8/src/builtins/builtins-api.cc:143
    #36 0x55e95454baa8 in ?? ??:0

0x62b000053b78 is located 0 bytes to the right of 27000-byte region [0x62b00004d200,0x62b000053b78)
allocated by thread T0 (chrome) here:
    #0 0x55e94c36533d in operator new[](unsigned long) /b/s/w/ir/cache/builder/src/third_party/llvm/compiler-rt/lib/asan/asan_new_delete.cpp:102
    #1 0x55e94c36533d in ?? ??:0
    #2 0x55e95a20ec4c in GrPixmap::Allocate(GrImageInfo const&) ./../../third_party/skia/src/gpu/GrPixmap.h:72
    #3 0x55e95a20ec4c in ?? ??:0
    #4 0x55e95a20ae7e in GrSurfaceContext::asyncRescaleAndReadPixelsYUV420(GrDirectContext*, SkYUVColorSpace, sk_sp<SkColorSpace>, SkIRect const&, SkISize,
SkImage::RescaleGamma, SkImage::RescaleMode, void (*)(void*, std::__1::unique_ptr<SkImage::AsyncReadResult const,
std::__1::default_delete<SkImage::AsyncReadResult const> >), void*) ./../../third_party/skia/src/gpu/GrSurfaceContext.cpp:964
    #5 0x55e95a20ae7e in ?? ??:0
    #6 0x55e95a4ffe3b in SkImage_Gpu::onAsyncRescaleAndReadPixelsYUV420(SkYUVColorSpace, sk_sp<SkColorSpace>, SkIRect const&, SkISize const&,
SkImage::RescaleGamma, SkImage::RescaleMode, void (*)(void*, std::__1::unique_ptr<SkImage::AsyncReadResult const,
std::__1::default_delete<SkImage::AsyncReadResult const> >), void*) ./../../third_party/skia/src/image/SkImage_Gpu.cpp:160
    #7 0x55e95a4ffe3b in ?? ??:0
    #8 0x55e94d06428d in SkImage::asyncRescaleAndReadPixelsYUV420(SkYUVColorSpace, sk_sp<SkColorSpace>, SkIRect const&, SkISize const&,
SkImage::RescaleGamma, SkImage::RescaleMode, void (*)(void*, std::__1::unique_ptr<SkImage::AsyncReadResult const,
std::__1::default_delete<SkImage::AsyncReadResult const> >), void*) ./../../third_party/skia/src/image/SkImage.cpp:96
    #9 0x55e94d06428d in ?? ??:0
    #10 0x55e96aa1e955 in blink::VideoFrame::Create(blink::ScriptState*, blink::ImageBitmap*, blink::VideoFrameInit*, blink::ExceptionState&)
./../../third_party/blink/renderer/modules/webcodecs/video_frame.cc:169
    #11 0x55e96aa1e955 in ?? ??:0
    #12 0x55e96aa2a3df in blink::(anonymous namespace)::ConstructorCallback(v8::FunctionCallbackInfo<v8::Value> const&)
./gen/third_party/blink/renderer/bindings/modules/v8/v8_video_frame.cc:270
    #13 0x55e96aa2a3df in ?? ??:0
    #14 0x55e954550331 in v8::internal::FunctionCallbackArguments::Call(v8::internal::CallHandlerInfo) ./../../v8/src/api/api-arguments-inl.h:158
    #15 0x55e954550331 in ?? ??:0
    #16 0x55e95454d0c6 in v8::internal::MaybeHandle<v8::internal::Object> v8::internal::(anonymous namespace)::HandleApiCallHelper<true>(v8::internal::Isolate*,
v8::internal::Handle<v8::internal::HeapObject>, v8::internal::Handle<v8::internal::HeapObject>, v8::internal::Handle<v8::internal::FunctionTemplateInfo>,
v8::internal::Handle<v8::internal::Object>, v8::internal::BuiltinArguments) ./../../v8/src/builtins/builtins-api.cc:113
    #17 0x55e95454d0c6 in ?? ??:0
    #18 0x55e95454ba1b in v8::internal::Builtin_Impl_HandleApiCall(v8::internal::BuiltinArguments, v8::internal::Isolate*) ./../../v8/src/builtins/builtins-api.cc:139
    #19 0x55e95454ba1b in ?? ??:0
    #20 0x55e9567488f7 in Builtins_CEntry_Return1_DontSaveFPRegs_ArgvOnStack_BuiltinExit setup-isolate-deserialize.cc:?
    #21 0x55e9567488f7 in ?? ??:0
    #22 0x55e9566dfb80 in Builtins_JSBuiltinsConstructStub setup-isolate-deserialize.cc:?
    #23 0x55e9566dfb80 in ?? ??:0
    #24 0x55e9567d4b1e in Builtins_ConstructHandler setup-isolate-deserialize.cc:?
    #25 0x55e9567d4b1e in ?? ??:0
    #26 0x55e9566e294e in Builtins_InterpreterEntryTrampoline setup-isolate-deserialize.cc:?
    #27 0x55e9566e294e in ?? ??:0
    #28 0x55e95679273a in Builtins_PromiseFulfillReactionJob setup-isolate-deserialize.cc:?
    #29 0x55e95679273a in ?? ??:0
    #30 0x55e956702a76 in Builtins_RunMicrotasks setup-isolate-deserialize.cc:?
    #31 0x55e956702a76 in ?? ??:0
    #32 0x55e9566e0517 in Builtins_JSRunMicrotasksEntry setup-isolate-deserialize.cc:?
    #33 0x55e9566e0517 in ?? ??:0
    #34 0x55e95480df88 in Call ./../../v8/src/execution/simulator.h:142
    #35 0x55e95480df88 in Invoke ./../../v8/src/execution/execution.cc:383
    #36 0x55e95480df88 in ?? ??:0
    #37 0x55e954811ab8 in v8::internal::(anonymous namespace)::InvokeWithTryCatch(v8::internal::Isolate*, v8::internal::(anonymous namespace)::InvokeParams const&)
./../../v8/src/execution/execution.cc:428
    #38 0x55e954811ab8 in ?? ??:0
    #39 0x55e954811f08 in v8::internal::Execution::TryRunMicrotasks(v8::internal::Isolate*, v8::internal::MicrotaskQueue*, v8::internal::MaybeHandle<v8::internal::Object>*)
./../../v8/src/execution/execution.cc:505
    #40 0x55e954811f08 in ?? ??:0
    #41 0x55e95489cad6 in v8::internal::MicrotaskQueue::RunMicrotasks(v8::internal::Isolate*) ./../../v8/src/execution/microtask-queue.cc:165
    #42 0x55e95489cad6 in ?? ??:0
    #43 0x55e95489c4b5 in v8::internal::MicrotaskQueue::PerformCheckpoint(v8::Isolate*) ./../../v8/src/execution/microtask-queue.cc:117
    #44 0x55e95489c4b5 in ?? ??:0
    #45 0x55e966d8f6ac in blink::V8ScriptRunner::RunCompiledScript(v8::Isolate*, v8::Local<v8::Script>, blink::ExecutionContext*)
./../../third_party/blink/renderer/bindings/core/v8/v8_script_runner.cc:372
    #46 0x55e966d8f6ac in ?? ??:0
    #47 0x55e966d908a9 in blink::V8ScriptRunner::CompileAndRunScript(v8::Isolate*, blink::ScriptState*, blink::ExecutionContext*, blink::ScriptSourceCode const&,
blink::KURL const&, blink::SanitizeScriptErrors, blink::ScriptFetchOptions const&, blink::ExecuteScriptPolicy, blink::V8ScriptRunner::RethrowErrorsOption)
./../../third_party/blink/renderer/bindings/core/v8/v8_script_runner.cc:462
    #48 0x55e966d908a9 in ?? ??:0
    #49 0x55e966cd344a in blink::ScriptController::ExecuteScriptAndReturnValue(v8::Local<v8::Context>, blink::ScriptSourceCode const&, blink::KURL const&,
blink::SanitizeScriptErrors, blink::ScriptFetchOptions const&, blink::ExecuteScriptPolicy) ./../../third_party/blink/renderer/bindings/core/v8/script_controller.cc:92
    #50 0x55e966cd344a in ?? ??:0
    #51 0x55e966cd5c5a in blink::ScriptController::EvaluateScriptInMainWorld(blink::ScriptSourceCode const&, blink::KURL const&, blink::SanitizeScriptErrors,
blink::ScriptFetchOptions const&, blink::ExecuteScriptPolicy) ./../../third_party/blink/renderer/bindings/core/v8/script_controller.cc:286
    #52 0x55e966cd5c5a in ?? ??:0
    #53 0x55e966679dd5 in RunScriptAndReturnValue ./../../third_party/blink/renderer/core/script/classic_script.cc:42
    #54 0x55e966679dd5 in RunScript ./../../third_party/blink/renderer/core/script/classic_script.cc:36
    #55 0x55e966679dd5 in RunScript ./../../third_party/blink/renderer/core/script/classic_script.cc:29
    #56 0x55e966679dd5 in ?? ??:0
    #57 0x55e9666ccc17 in blink::PendingScript::ExecuteScriptBlockInternal(blink::Script*, blink::ScriptElementBase*, bool, bool, bool, base::TimeTicks, bool)
./../../third_party/blink/renderer/core/script/pending_script.cc:264
    #58 0x55e9666ccc17 in ?? ??:0

#59 0x55e9666cc531 in blink::PendingScript::ExecuteScriptBlock(blink::KURL const&) ./../../third_party/blink/renderer/core/script/pending_script.cc:170
#60 0x55e9666cc531 in ?? ??:0
#61 0x55e9666c36fa in blink::ScriptLoader::PrepareScript(WTF::TextPosition const&, blink::ScriptLoader::LegacyTypeSupport)
./../../third_party/blink/renderer/core/script/script_loader.cc:960
#62 0x55e9666c36fa in ?? ??:0

Did this work before? N/A

Chrome version: Chromium 89.0.4381.6 Channel: dev
OS Version: 20.04
Flash Version:

crash.html
322 bytes View Download

asan-89.0.4388.0.log
13.8 KB View Download

Comment 1 by sheriffbot on Thu, Jan 14, 2021, 1:11 AM EST

Labels: reward-potential

Comment 2 by ClusterFuzz on Thu, Jan 14, 2021, 3:16 PM EST

ClusterFuzz is analyzing your testcase. Developers can follow the progress at https://clusterfuzz.com/testcase?key=5183233699676160.

Comment 3 by ClusterFuzz on Thu, Jan 14, 2021, 8:09 PM EST

Labels: Unreproducible

ClusterFuzz testcase 5183233699676160 appears to be flaky, updating reproducibility label.

Comment 4 by ClusterFuzz on Thu, Jan 14, 2021, 8:09 PM EST

Detailed Report: https://clusterfuzz.com/testcase?key=5183233699676160

Fuzzer: None
Job Type: linux_asan_chrome_mp
Platform Id: linux

Crash Type:
Crash Address:
Crash State:

Sanitizer: address (ASAN)

Crash Revision: https://clusterfuzz.com/revisions?job=linux_asan_chrome_mp&revision=843610

Reproducer Testcase: https://clusterfuzz.com/download?testcase_id=5183233699676160

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

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

To reproduce this issue, run:

./reproduce.sh -t https://clusterfuzz.com/testcase-detail/5183233699676160 -b /path/to/build

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

********** UNREPRODUCIBLE **********
Note: This crash might not be reproducible with the provided testcase. That said, for the past 14 days, we've been seeing this crash frequently.

It may be possible to reproduce by trying the following options:
- Run testcase multiple times for a longer duration.
- Run fuzzing without testcase argument to hit the same crash signature.

If it still does not reproduce, try a speculative fix based on the crash stacktrace and verify if it works by looking at the crash statistics in the report. We will auto-close the bug if the crash is not seen for 14 days.
****

Comment 5 by xinghuilu@chromium.org on Fri, Jan 15, 2021, 1:50 PM EST

Status: Assigned (was: Unconfirmed)
Owner: nazab...@microsoft.com
Cc: sande...@chromium.org
Labels: Security_Impact-Stable Security_Severity-High
Components: Internals>Media

Thanks for the report. This is likely introduced in https://crrev.com/c/2363021. nazabris@, could you take a look? Thanks!

Comment 6 by adetaylor@google.com on Fri, Jan 15, 2021, 1:56 PM EST

Labels: external_security_report

Comment 7 by sheriffbot on Sat, Jan 16, 2021, 12:47 PM EST

Labels: M-87 Target-87

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

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

Comment 8 by sheriffbot on Sat, Jan 16, 2021, 1:27 PM EST

Labels: -Pri-2 Pri-1

Setting Pri-1 to match security severity High. If this is incorrect, please reset the priority. Sheriffbot won't make this change again.

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

Comment 9 by sheriffbot on Wed, Jan 20, 2021, 12:21 PM EST

Labels: -M-87 Target-88 M-88

Comment 10 by adetaylor@google.com on Wed, Jan 20, 2021, 7:01 PM EST

Labels: -reward-potential

Comment 11 by bugdroid on Fri, Jan 22, 2021, 3:51 PM EST

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08

commit 7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08
Author: Nathan Zabriskie <nazabris@microsoft.com>
Date: Fri Jan 22 20:50:41 2021

Fix heap overflow in VideoFrameYUVConverter

Currently with some texture sizes GLES2Util::ComputeImageDataSizesES3
will attempt to add row padding when calculating the size of a
VideoFrame plane. This is because it's currently assumed that each row
aligns on a 4 byte boundary based on GL_UNPACK_ALIGNMENT but
VideoFrames make no such guarantee as they may be densely packed.
This CL removes the GL_UNPACK_ALIGNMENT assumption so that we only use
the VideoFrame's stride when calculating padding.

~~Bug: 1166504~~, ~~1161131~~
Change-Id: I2484f5dfd2ad85b088fee57758776a5c9bd01d95
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2642765
Reviewed-by: Vasiliy Telezhnikov <vasilyt@chromium.org>
Commit-Queue: Nathan Zabriskie <nazabris@microsoft.com>
Cr-Commit-Position: refs/heads/master@{#846298}

[modify] https://crrev.com/7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08/gpu/command_buffer/client/raster_implementation_gles.cc
[modify] https://crrev.com/7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08/gpu/command_buffer/client/gles2_implementation.cc

Comment 12 Deleted

Comment 13 by nazab...@microsoft.com on Fri, Jan 22, 2021, 3:54 PM EST

This issue no longer repros for me locally with the change in #11. Do we need to cherry pick this change into any other branches?

Comment 14 by nazab...@microsoft.com on Mon, Jan 25, 2021, 2:06 PM EST

Cc: vasilyt@chromium.org

Comment 15 by nazab...@microsoft.com on Tue, Jan 26, 2021, 2:50 PM EST

Labels: Merge-Request-89

Do we need this fix in any release branches due to security label?

Comment 16 by adetaylor@google.com on Tue, Jan 26, 2021, 7:48 PM EST

Yes. Is #c11 a complete fix? If so please mark the bug as fixed, then merge processes will kick off.

Comment 17 by sheriffbot on Wed, Jan 27, 2021, 2:55 PM EST

Labels: -Merge-Request-89 Hotlist-Merge-Approved Merge-Approved-89

Your change meets the bar and is auto-approved for M89. Please go ahead and merge the CL to branch 4389 (refs/branch-heads/4389) manually. Please contact milestone owner if you have questions.
Merge instructions: https://www.chromium.org/developers/how-tos/drover
Owners: benmason@(Android), bindusuvarna@(iOS), geohsu@(ChromeOS), pbommana@(Desktop)

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

Comment 18 by nazab...@microsoft.com on Wed, Jan 27, 2021, 3:03 PM EST

Status: Fixed (was: Assigned)

Comment 19 by nazab...@microsoft.com on Wed, Jan 27, 2021, 3:14 PM EST

Cc: adetaylor@google.com

Yes #c11 is a complete fix. I marked the bug as fixed and followed the instructions in #c17 but the rubber stamp bot did not approve the change. Who should I add to the patch for merge approval? Thanks!

Comment 20 by adetaylor@google.com on Wed, Jan 27, 2021, 3:29 PM EST

I don't think the bot approves merge CLs, it just adds merge approvals here. Please arrange for someone involved in the original fix to approve the merge CL, e.g. vasilyt@?

Comment 21 by vasilyt@chromium.org on Wed, Jan 27, 2021, 3:45 PM EST

Cc: pbomm...@chromium.org

Rubber stamp bot should CR+1 clean merges, but other people reported similar problems too (see Issue 1092608).
lgtmed.

+cc pbommana to avoid confusion, as the merge of this CL was requested in two different bugs.

Comment 22 by bugdroid on Wed, Jan 27, 2021, 5:31 PM EST

Labels: -merge-approved-89 merge-merged-89 merge-merged-4389

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/0ea74719c2da1d6a670e8f172fb6ed8ed04d01f6

commit 0ea74719c2da1d6a670e8f172fb6ed8ed04d01f6
Author: Nathan Zabriskie <nazabris@microsoft.com>
Date: Wed Jan 27 22:21:25 2021

Fix heap overflow in VideoFrameYUVConverter

Currently with some texture sizes GLES2Util::ComputeImageDataSizesES3
will attempt to add row padding when calculating the size of a
VideoFrame plane. This is because it's currently assumed that each row
aligns on a 4 byte boundary based on GL_UNPACK_ALIGNMENT but
VideoFrames make no such guarantee as they may be densely packed.
This CL removes the GL_UNPACK_ALIGNMENT assumption so that we only use
the VideoFrame's stride when calculating padding.

(cherry picked from commit 7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08)

~~Bug: 1166504~~, ~~1161131~~
Change-Id: I2484f5dfd2ad85b088fee57758776a5c9bd01d95
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2642765
Reviewed-by: Vasiliy Telezhnikov <vasilyt@chromium.org>
Commit-Queue: Nathan Zabriskie <nazabris@microsoft.com>

Cr-Original-Commit-Position: refs/heads/master@{#846298}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2648207
Auto-Submit: Nathan Zabriskie <nazabris@microsoft.com>
Commit-Queue: Vasiliy Telezhnikov <vasilyt@chromium.org>
Cr-Commit-Position: refs/branch-heads/4389@{#326}
Cr-Branched-From: 9251c5db2b6d5a59fe4eac7aafa5fed37c139bb7-refs/heads/master@{#843830}

[modify] https://crrev.com/0ea74719c2da1d6a670e8f172fb6ed8ed04d01f6/gpu/command_buffer/client/raster_implementation_gles.cc
[modify] https://crrev.com/0ea74719c2da1d6a670e8f172fb6ed8ed04d01f6/gpu/command_buffer/client/gles2_implementation.cc

Comment 23 by sheriffbot on Thu, Jan 28, 2021, 12:40 PM EST

Labels: reward-topanel

Comment 24 by sheriffbot on Thu, Jan 28, 2021, 1:55 PM EST

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

Comment 25 by adetaylor@google.com on Fri, Jan 29, 2021, 4:56 PM EST

Labels: Merge-Request-88

Unfortunately sheriffbot thought #c15 was outsmarting its robot peanut brain, so it didn't ask for merge request to 88. We should at least consider it.

Comment 26 by adetaylor@google.com on Fri, Jan 29, 2021, 5:00 PM EST

This didn't quite make it into the M88 release that will go out on Tuesday, but after that's gone I will approve merge for the one after.

Comment 27 by nazab...@microsoft.com on Fri, Feb 5, 2021, 12:54 PM EST

Should we merge the fix to 88 now that the Tuesday release has passed?

Comment 28 by adetaylor@google.com on Fri, Feb 5, 2021, 1:31 PM EST

Labels: -Merge-Request-88 Merge-Approved-88

Yep. I usually do a batch of merge approvals a few days before the next release is cut. But happy to approve this one now - approving merge to M88, branch 4324.

Comment 29 by nazab...@microsoft.com on Fri, Feb 5, 2021, 1:42 PM EST

Ah I'll go ahead and merge it now but I'll keep that in mind for the future :) Thanks!

Comment 30 by bugdroid on Fri, Feb 5, 2021, 3:45 PM EST

Labels: -merge-approved-88 merge-merged-4324 merge-merged-88

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/59f3ca278089f630613bf8a50e7711244dfce5fd

commit 59f3ca278089f630613bf8a50e7711244dfce5fd
Author: Nathan Zabriskie <nazabris@microsoft.com>
Date: Fri Feb 05 20:44:16 2021

Fix heap overflow in VideoFrameYUVConverter

Currently with some texture sizes GLES2Util::ComputeImageDataSizesES3
will attempt to add row padding when calculating the size of a
VideoFrame plane. This is because it's currently assumed that each row
aligns on a 4 byte boundary based on GL_UNPACK_ALIGNMENT but
VideoFrames make no such guarantee as they may be densely packed.
This CL removes the GL_UNPACK_ALIGNMENT assumption so that we only use
the VideoFrame's stride when calculating padding.

(cherry picked from commit 7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08)

~~Bug: 1166504~~, ~~1164131~~
Change-Id: I2484f5dfd2ad85b088fee57758776a5c9bd01d95
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2642765
Reviewed-by: Vasiliy Telezhnikov <vasilyt@chromium.org>
Commit-Queue: Nathan Zabriskie <nazabris@microsoft.com>
Cr-Original-Commit-Position: refs/heads/master@{#846298}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2679121
Bot-Commit: Rubber Stamper <rubber-stamper@appspot.gserviceaccount.com>
Auto-Submit: Nathan Zabriskie <nazabris@microsoft.com>
Cr-Commit-Position: refs/branch-heads/4324@{#2115}
Cr-Branched-From: c73b5a651d37a6c4d0b8e3262cc4015a5579c6c8-refs/heads/master@{#827102}

[modify] https://crrev.com/59f3ca278089f630613bf8a50e7711244dfce5fd/gpu/command_buffer/client/raster_implementation_gles.cc
[modify] https://crrev.com/59f3ca278089f630613bf8a50e7711244dfce5fd/gpu/command_buffer/client/gles2_implementation.cc

Comment 31 by amyressler@google.com on Wed, Feb 10, 2021, 1:59 PM EST

Labels: -reward-topanel reward-unpaid reward-5000

*** 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 32 by amyressler@google.com on Wed, Feb 10, 2021, 5:50 PM EST

Congratulations! The VRP Panel has decided to award you $5,000 for this report. Nice work!

Comment 33 by amyressler@google.com on Thu, Feb 11, 2021, 4:00 PM EST

Labels: -reward-unpaid reward-inprocess

Comment 34 by adetaylor@google.com on Fri, Feb 12, 2021, 7:35 PM EST

Labels: Release-3-M88

Comment 35 by achuith@chromium.org on Thu, Feb 18, 2021, 8:56 PM EST

Labels: LTS-Security-86 Merge-Request-86-LTS

Comment 36 by amyressler@google.com on Mon, Feb 22, 2021, 4:31 PM EST

Labels: CVE-2021-21152 CVE_description-missing

Comment 37 by amyressler@google.com on Mon, Feb 22, 2021, 4:33 PM EST

Labels: -CVE_description-missing CVE_description-submitted

Comment 38 by gianluca@google.com on Tue, Feb 23, 2021, 4:31 PM EST

Labels: -Merge-Request-86-LTS LTS-Merge-Request-86

Comment 39 by gianluca@google.com on Tue, Feb 23, 2021, 5:18 PM EST

Labels: LTS-Merge-Approved-86

Comment 40 by achuith@chromium.org on Tue, Feb 23, 2021, 5:30 PM EST

Labels: -LTS-Merge-Request-86

Comment 41 by bugdroid on Tue, Feb 23, 2021, 6:47 PM EST

Labels: merge-merged-4240 merge-merged-86

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/cc2480d04a3e719f5a882d96a0a50f574be4d653

commit cc2480d04a3e719f5a882d96a0a50f574be4d653
Author: Nathan Zabriskie <nazabris@microsoft.com>
Date: Tue Feb 23 23:46:23 2021

Fix heap overflow in VideoFrameYUVConverter

Currently with some texture sizes GLES2Util::ComputeImageDataSizesES3
will attempt to add row padding when calculating the size of a
VideoFrame plane. This is because it's currently assumed that each row
aligns on a 4 byte boundary based on GL_UNPACK_ALIGNMENT but
VideoFrames make no such guarantee as they may be densely packed.
This CL removes the GL_UNPACK_ALIGNMENT assumption so that we only use
the VideoFrame's stride when calculating padding.

(cherry picked from commit 7de5d0ecb5a4f73aeffe15d825bf694d0d8e2a08)

~~Bug: 1166504~~, ~~1161131~~
Change-Id: I2484f5dfd2ad85b088fee57758776a5c9bd01d95
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2642765
Reviewed-by: Vasiliy Telezhnikov <vasilyt@chromium.org>
Commit-Queue: Nathan Zabriskie <nazabris@microsoft.com>
Cr-Original-Commit-Position: refs/heads/master@{#846298}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2706228
Reviewed-by: Victor-Gabriel Savu <vsavu@google.com>
Commit-Queue: Achuith Bhandarkar <achuith@chromium.org>
Cr-Commit-Position: refs/branch-heads/4240@{#1547}
Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify] https://crrev.com/cc2480d04a3e719f5a882d96a0a50f574be4d653/gpu/command_buffer/client/raster_implementation_gles.cc
[modify] https://crrev.com/cc2480d04a3e719f5a882d96a0a50f574be4d653/gpu/command_buffer/client/gles2_implementation.cc

Comment 42 by asumaneev@google.com on Tue, Mar 2, 2021, 10:33 AM EST

Labels: -LTS-Merge-Approved-86 LTR-Merged-86

Comment 43 by sheriffbot on Thu, May 6, 2021, 1:51 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@chromium.org on Wed, Jan 5, 2022, 4:43 PM EST

Hello OP/emilykim@, we consider attachments/pocs included with reports to be an integral part of the report (https://bughunters.google.com/about/rules/5745167867576320), so I've undeleted them. Thank you!