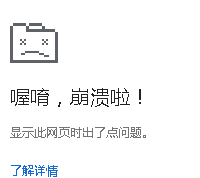
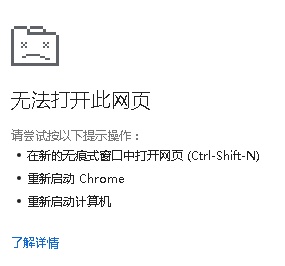
# GrOp::gCurrOpUniqueID溢出导致Chrome进程崩溃

## 背景

公司的大屏展示，利用Canvas做了较多2D图像绘制，客户处发现，当大屏持续开启一个周左右，Chrome会崩溃，崩溃UI类似：



我们从客户的Chrome://crashes里面获取到了崩溃的dump文件（位于：C:\Users\你的用户名\AppData\Local\Google\Chrome\User Data\Crashpad\reports）（注：下面的截图是我们自己的截图，不是客户的）。



## 原因分析

由于网络原因，之前一直没有下载到对应的Chrome符号，昨晚第二次分析，发现符号可以下载了，于是使用Windbg，加入symbols路径：

SRV\*C:\Symbols\*http://msdl.microsoft.com/download/symbols;SRV\*C:\Symbols\*https://chromium-browser-symsrv.commondatastorage.googleapis.com

1. **先看下analyze结果**

Executable search path is:

Windows 7 Version 17134 (1) MP (4 procs) Free x64

Product: WinNt

Machine Name:

Debug session time: Sat Nov 24 20:32:02.000 2018 (UTC + 8:00) //周六晚上崩溃的

System Uptime: not available

Process Uptime: 2 days 11:52:48.000

0:000> !analyze -v

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* Exception Analysis \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Failed calling InternetOpenUrl, GLE=12002

FAULTING\_IP:

chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort+0 [C:\b\c\b\win64\_clang\src\base\win\win\_util.cc @ 87]

00007ff9`48b9cd80 c704250000000037130000 mov dword ptr [0],1337h // 这是一段强制Abort代码，访问空指针。

EXCEPTION\_RECORD: ffffffffffffffff -- (.exr 0xffffffffffffffff)

ExceptionAddress: 00007ff948b9cd80 (chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort)

ExceptionCode: c0000005 (Access violation)

ExceptionFlags: 00000000

NumberParameters: 2

Parameter[0]: 0000000000000001

Parameter[1]: 0000000000000000

Attempt to write to address 0000000000000000

DEFAULT\_BUCKET\_ID: NULL\_POINTER\_WRITE

PROCESS\_NAME: chrome.exe

ERROR\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_PARAMETER1: 0000000000000001

EXCEPTION\_PARAMETER2: 0000000000000000

WRITE\_ADDRESS: 0000000000000000

FOLLOWUP\_IP:

chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort+0 [C:\b\c\b\win64\_clang\src\base\win\win\_util.cc @ 87]

00007ff9`48b9cd80 c704250000000037130000 mov dword ptr [0],1337h

NTGLOBALFLAG: 0

FAULTING\_THREAD: 00000000000016cc

PRIMARY\_PROBLEM\_CLASS: NULL\_POINTER\_WRITE

BUGCHECK\_STR: APPLICATION\_FAULT\_NULL\_POINTER\_WRITE

LAST\_CONTROL\_TRANSFER: from 00007ff94b10f6e7 to 00007ff948b9cd80

STACK\_TEXT:

000000bc`c85fca78 00007ff9`4b10f6e7 : 00007ff9`4bf74c98 00007ff9`4b10f857 000000bc`c85fcaf8 00000000`00000201 : chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort [C:\b\c\b\win64\_clang\src\base\win\win\_util.cc @ 87]

000000bc`c85fca80 00007ff9`4b0fe654 : 00000000`00000101 0000027a`00000000 00000000`00000000 0000027a`c7be07c8 : chrome\_child!raise+0x22b [minkernel\crts\ucrt\src\appcrt\misc\signal.cpp @ 547]

000000bc`c85fcaf0 00007ff9`48d2c5c5 : 0000027a`c7be07c8 0000027a`c91d4af8 00000000`00000000 000000bc`c85fcc60 : chrome\_child!abort+0x18 [minkernel\crts\ucrt\src\appcrt\startup\abort.cpp @ 71]

000000bc`c85fcb20 00007ff9`47995d44 : 43978309`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!sk\_abort\_no\_print+0x15 [C:\b\c\b\win64\_clang\src\skia\ext\SkMemory\_new\_handler.cpp @ 41]

000000bc`c85fcb50 00007ff9`4a111464 : 0000027a`b3f31650 00007ff9`48f5e14f 00007ff9`4b432658 00007ff9`4b432658 : chrome\_child!GrOpFlushState::draw+0x1b4 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpFlushState.cpp @ 120] // 这里是abort调用之前的代码。

000000bc`c85fcbe0 00007ff9`47990cf7 : 0000027a`be6fdca8 0000027a`be6fdca8 00000000`00000005 000000bc`c85fd540 : chrome\_child!`anonymous namespace'::AAConvexPathOp::onPrepareDraws+0x2754 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\ops\GrAAConvexPathRenderer.cpp @ 943]

000000bc`c85fd110 00007ff9`47990b41 : 0000027a`c9406730 000000bc`c85fd4b0 000000bc`c85fd304 000000bc`c85fd4b0 : chrome\_child!GrRenderTargetOpList::onPrepare+0x197 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrRenderTargetOpList.cpp @ 105]

000000bc`c85fd1b0 00007ff9`47990830 : 0000027a`bac81dd0 0000027a`bac81e18 00000000`000000ff 00000000`00000006 : chrome\_child!GrOpList::prepare+0x61 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpList.cpp @ 88]

000000bc`c85fd210 00007ff9`4798f4cf : 00000000`00000000 00000000`00000000 00000000`00000000 00005c10`68089d0f : chrome\_child!GrDrawingManager::executeOpLists+0xb0 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrDrawingManager.cpp @ 315]

000000bc`c85fd280 00007ff9`47bda0d7 : 00007ff9`4b299e07 00005c10`6808981f 00007ff9`4b299e07 00007ff9`4b299dfe : chrome\_child!GrDrawingManager::internalFlush+0x75f [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrDrawingManager.cpp @ 232]

000000bc`c85fd7a0 00007ff9`48d896b4 : 000000bc`c85fd830 0000027a`8127fd18 0000027a`c8b5fef0 00007ff9`4b3a2d30 : chrome\_child!GrContext::flush+0x27 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrContext.cpp @ 366]

000000bc`c85fd7d0 00007ff9`4798be1b : 0000027a`b40b0000 00007ff9`917b2b55 0000027a`31000031 0000027a`00000002 : chrome\_child!cc::SkiaPaintCanvas::FlushAfterDrawIfNeeded+0x74 [C:\b\c\b\win64\_clang\src\cc\paint\skia\_paint\_canvas.cc @ 376]

000000bc`c85fd8b0 00007ff9`48d8a3bd : 0000027a`c84e89e0 00007ff9`475d2bb5 000000bc`c85fdce8 00007ff9`4b0c343b : chrome\_child!cc::PaintOpBuffer::Playback+0x20b [C:\b\c\b\win64\_clang\src\cc\paint\paint\_op\_buffer.cc @ 2333]

000000bc`c85fdb20 00007ff9`48d8a2c1 : 00000000`00000000 00000000`00000000 00005c10`6808958f 00005c10`680894ef : chrome\_child!cc::SkiaPaintCanvas::drawPicture+0xcd [C:\b\c\b\win64\_clang\src\cc\paint\skia\_paint\_canvas.cc @ 364]

000000bc`c85fdbe0 00007ff9`4a298fd8 : 00007ff9`4b299e07 00007ff9`4b299dfe 00000000`00000000 00000000`00000000 : chrome\_child!cc::SkiaPaintCanvas::drawPicture+0x41 [C:\b\c\b\win64\_clang\src\cc\paint\skia\_paint\_canvas.cc @ 321]

000000bc`c85fdc30 00007ff9`4a299e67 : 00000000`00000000 00005c10`680893ff 00005c10`6808938f 00000000`00000000 : chrome\_child!blink::Canvas2DLayerBridge::FlushRecording+0xb4 [C:\b\c\b\win64\_clang\src\third\_party\blink\renderer\platform\graphics\canvas\_2d\_layer\_bridge.cc @ 517]

000000bc`c85fdd20 00007ff9`492e5695 : 0000027a`b671c8f0 00007ff9`47493fcc 00000000`00000000 00000000`00000000 : chrome\_child!blink::Canvas2DLayerBridge::PrepareTransferableResource+0x5d [C:\b\c\b\win64\_clang\src\third\_party\blink\renderer\platform\graphics\canvas\_2d\_layer\_bridge.cc @ 615]

000000bc`c85fdd80 00007ff9`477d1c6a : 00000000`00000000 00000000`00000000 00000003`c85fde00 00000000`00000000 : chrome\_child!cc::TextureLayer::Update+0x65 [C:\b\c\b\win64\_clang\src\cc\layers\texture\_layer.cc @ 179]

000000bc`c85fde80 00007ff9`477d0f71 : 0000027a`c8cfd300 00005c10`6808aedf 00000000`00000002 000000bc`c85fe520 : chrome\_child!cc::LayerTreeHost::DoUpdateLayers+0xc2a [C:\b\c\b\win64\_clang\src\cc\trees\layer\_tree\_host.cc @ 820]

000000bc`c85fe2a0 00007ff9`477c44a3 : 0000027a`b3ef1358 0000027a`b3ef1700 0000027a`c7ac6f48 0000027a`c7ac6ec0 : chrome\_child!cc::LayerTreeHost::UpdateLayers+0x41 [C:\b\c\b\win64\_clang\src\cc\trees\layer\_tree\_host.cc @ 647]

000000bc`c85fe320 00007ff9`477c4076 : 00000000`00000002 0000027a`b3ee1c60 000000bc`c85fe7c0 00007ff9`473b86e8 : chrome\_child!cc::ProxyMain::BeginMainFrame+0x403 [C:\b\c\b\win64\_clang\src\cc\trees\proxy\_main.cc @ 270]

000000bc`c85fe4f0 00007ff9`473b851c : 0000027a`b3ef1708 00007ff9`4b87ee9a 00000000`00000001 0000027a`b3ef1700 : chrome\_child!base::internal::Invoker<base::internal::BindState<void (cc::ProxyMain::\*)(std::unique\_ptr<cc::BeginMainFrameAndCommitState,std::default\_delete<cc::BeginMainFrameAndCommitState> >),base::WeakPtr<cc::ProxyMain>,base::internal::PassedWrapper<std::unique\_ptr<cc::BeginMainFrameAndCommitState,std::default\_delete<cc::BeginMainFrameAndCommitState> > > >,void ()>::RunOnce+0xa6 [C:\b\c\b\win64\_clang\src\base\bind\_internal.h @ 662]

000000bc`c85fe560 00007ff9`47414996 : 00000000`00000000 0000027a`b3ee1bc0 000000bc`c85fe7c8 00007ff9`47419275 : chrome\_child!base::debug::TaskAnnotator::RunTask+0x12c [C:\b\c\b\win64\_clang\src\base\debug\task\_annotator.cc @ 101]

000000bc`c85fe680 00007ff9`473b851c : 00000000`00000000 00005c10`6808a83f 00000000`00000000 00000000`00000000 : chrome\_child!base::sequence\_manager::internal::ThreadControllerImpl::DoWork+0x1b6 [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\thread\_controller\_impl.cc @ 179]

000000bc`c85fe8c0 00007ff9`473b7d7f : 00005c10`6808a69f 00000000`b3ee1a01 000000bc`c85ff101 000000bc`c85feaa8 : chrome\_child!base::debug::TaskAnnotator::RunTask+0x12c [C:\b\c\b\win64\_clang\src\base\debug\task\_annotator.cc @ 101]

000000bc`c85fe9e0 00007ff9`473b1215 : 000000bc`c85fecb0 00007ff9`473b0436 0000027a`b3ee1b20 00007ff9`47414e02 : chrome\_child!base::MessageLoop::RunTask+0xdf [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 436]

000000bc`c85feb20 00007ff9`473b1069 : 000000bc`c85feef0 00007ff9`47393d74 000000bc`c85fee68 00007ff9`4bdde830 : chrome\_child!base::MessageLoop::DoWork+0x185 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 517]

000000bc`c85fed50 00007ff9`473b05f1 : 00000000`00000000 00000000`b71d5e01 00007ff9`4bd77780 000000bc`c85fee68 : chrome\_child!base::MessagePumpDefault::Run+0x99 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_pump\_default.cc @ 37]

000000bc`c85fedb0 00007ff9`473939cc : 0000027a`b40b0340 000000bc`c85fee98 00000000`00000000 000000bc`c85feeb0 : chrome\_child!base::RunLoop::Run+0x31 [C:\b\c\b\win64\_clang\src\base\run\_loop.cc @ 108]

000000bc`c85fede0 00007ff9`4738d2e3 : 000000bc`c85ff080 00007ff9`4b0c343b 000000bc`c85ff080 00000000`00000003 : chrome\_child!content::RendererMain+0x3d6 [C:\b\c\b\win64\_clang\src\content\renderer\renderer\_main.cc @ 200]

000000bc`c85fefa0 00007ff9`47364968 : 00000000`00000000 00000000`00000027 00000000`00000000 00000000`00000000 : chrome\_child!content::ContentMainRunnerImpl::Run+0x171 [C:\b\c\b\win64\_clang\src\content\app\content\_main\_runner\_impl.cc @ 898]

000000bc`c85ff140 00007ff9`47364568 : 000000bc`c85ff6a8 00007ff9`47361d5b 00000000`000001f6 00007ff9`4736111b : chrome\_child!service\_manager::Main+0x333 [C:\b\c\b\win64\_clang\src\services\service\_manager\embedder\main.cc @ 472]

000000bc`c85ff4b0 00007ff9`473619f8 : 000000bc`c85ff560 00007ff6`b70a0000 000000bc`c85ff680 00000000`00000000 : chrome\_child!content::ContentMain+0x41 [C:\b\c\b\win64\_clang\src\content\app\content\_main.cc @ 19]

000000bc`c85ff540 00007ff6`b70a376c : 0000027a`b3b88a60 00007ff9`473618e0 000000bc`c85ff6a8 0000027a`b3b88a50 : chrome\_child!ChromeMain+0x118 [C:\b\c\b\win64\_clang\src\chrome\app\chrome\_main.cc @ 0]

000000bc`c85ff620 00007ff6`b70a1697 : 00000000`00000018 ffffffff`fffffffe 0000027a`b3b8030c 00000000`0000001a : chrome!MainDllLoader::Launch+0x28c [C:\b\c\b\win64\_clang\src\chrome\app\main\_dll\_loader\_win.cc @ 205]

000000bc`c85ff710 00000000`00000000 : 00007ff6`e2000000 00000000`00000000 72657265`646e6572 0000027a`b3b83400 : chrome!std::operator<<<std::char\_traits<char> >+0x47 [C:\b\c\b\win64\_clang\src\third\_party\depot\_tools\win\_toolchain\vs\_files\3bc0ec615cf20ee342f3bc29bc991b5ad66d8d2c\VC\Tools\MSVC\14.14.26428\include\ostream @ 787]

STACK\_COMMAND: ~0s; .ecxr ; kb

SYMBOL\_STACK\_INDEX: 0

SYMBOL\_NAME: chrome!base::win::`anonymous namespace'::ForceCrashOnSigAbort+0

FOLLOWUP\_NAME: MachineOwner

MODULE\_NAME: chrome\_child

IMAGE\_NAME: chrome\_child.dll

DEBUG\_FLR\_IMAGE\_TIMESTAMP: 5becfd50

FAILURE\_BUCKET\_ID: NULL\_POINTER\_WRITE\_c0000005\_chrome\_child.dll!base::win::\_anonymous\_namespace\_::ForceCrashOnSigAbort

BUCKET\_ID: X64\_APPLICATION\_FAULT\_NULL\_POINTER\_WRITE\_chrome!base::win::\_anonymous\_namespace\_::ForceCrashOnSigAbort+0

WATSON\_STAGEONE\_URL: http://watson.microsoft.com/StageOne/chrome\_exe/70\_0\_3538\_110/5becfd50/chrome\_child\_dll/70\_0\_3538\_110/5becfd50/c0000005/0183cd80.htm?Retriage=1

Followup: MachineOwner

---------

0:000> lmvm chrome\_child

start end module name

00007ff9`47360000 00007ff9`4c253000 chrome\_child C (private pdb symbols) c:\symbols\chrome\_child.dll.pdb\61754B485B7E8429FE4BA7DC749B3ECC1\chrome\_child.dll.pdb

Loaded symbol image file: chrome\_child.dll

Mapped memory image file: c:\symbols\chrome\_child.dll\5BECFD504ef3000\chrome\_child.dll

Image path: D:\Google\Chrome\Application\70.0.3538.110\chrome\_child.dll

Image name: chrome\_child.dll

Timestamp: Thu Nov 15 13:00:00 2018 (5BECFD50)

CheckSum: 00000000

ImageSize: 04EF3000

File version: 70.0.3538.110

Product version: 70.0.3538.110

File flags: 0 (Mask 0)

File OS: 0 Unknown Base

File type: 1.0 App

File date: 00000000.00000000

Translations: 0409.04b0

CompanyName: Google Inc.

ProductName: Google Chrome

InternalName: chrome\_dll

OriginalFilename: chrome.dll

ProductVersion: 70.0.3538.110 //这是客户的Chrome版本

FileVersion: 70.0.3538.110

FileDescription: Google Chrome

LegalCopyright: Copyright 2017 Google Inc. All rights reserved.

0:000> ~\*kv

. 0 Id: 1704.16cc Suspend: 0 Teb: 000000bc`c83e2000 Unfrozen

Child-SP RetAddr : Args to Child : Call Site

000000bc`c85fb7e8 00007ff9`8de25e9a : 000000bc`c85fb8a8 00000000`00000000 00000000`00000000 00000000`00000000 : ntdll!ZwDelayExecution+0x14

000000bc`c85fb7f0 00007ff9`7d5d2e15 : 000000bc`c85fb998 00001b18`00000000 00000000`c00000bb 00000000`00000000 : KERNELBASE!SleepEx+0x9a

000000bc`c85fb890 00007ff9`8decbcf0 : 00000000`00000000 000000bc`c85ffb60 00000000`00000000 00007ff9`917b5c52 : chrome\_elf!crashpad::`anonymous namespace'::UnhandledExceptionHandler+0xdd [C:\b\c\b\win64\_clang\src\third\_party\crashpad\crashpad\client\crashpad\_client\_win.cc @ 174]

000000bc`c85fb9e0 00007ff9`91842757 : 00007ff6`b71ceb60 00007ff9`918dc15c 00000000`00000000 00000000`00000000 : KERNELBASE!UnhandledExceptionFilter+0x190

000000bc`c85fbaf0 00007ff9`9182ab46 : 000000bc`c85fbb89 0000027a`b40b0150 00000000`00000000 000000bc`c85fc7b0 : ntdll!RtlUserThreadStart$filt$0+0x38

000000bc`c85fbb20 00007ff9`9183ed3d : 00000000`00000000 000000bc`c85fbcc0 000000bc`c85fc2c0 000000bc`c85fbcc0 : ntdll!\_C\_specific\_handler+0x96

000000bc`c85fbb90 00007ff9`917a6c86 : 000000bc`c85fbcc0 000000bc`c85fc2c0 00007ff9`8ee7ab10 00000000`00000000 : ntdll!RtlpExecuteHandlerForException+0xd

000000bc`c85fbbc0 00007ff9`9183dc6e : 00000000`000000d0 435fd7c3`447a34e2 00000000`00000016 00000000`000000d0 : ntdll!RtlDispatchException+0x3c6

000000bc`c85fc2c0 00007ff9`48b9cd80 : 00007ff9`4b10f6e7 00007ff9`4bf74c98 00007ff9`4b10f857 000000bc`c85fcaf8 : ntdll!KiUserExceptionDispatcher+0x2e (TrapFrame @ 000000bc`c85fc6e8)

000000bc`c85fca78 00007ff9`4b10f6e7 : 00007ff9`4bf74c98 00007ff9`4b10f857 000000bc`c85fcaf8 00000000`00000201 : chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort [C:\b\c\b\win64\_clang\src\base\win\win\_util.cc @ 87]

000000bc`c85fca80 00007ff9`4b0fe654 : 00000000`00000101 0000027a`00000000 00000000`00000000 0000027a`c7be07c8 : chrome\_child!raise+0x22b [minkernel\crts\ucrt\src\appcrt\misc\signal.cpp @ 547]

000000bc`c85fcaf0 00007ff9`48d2c5c5 : 0000027a`c7be07c8 0000027a`c91d4af8 00000000`00000000 000000bc`c85fcc60 : chrome\_child!abort+0x18 [minkernel\crts\ucrt\src\appcrt\startup\abort.cpp @ 71]

000000bc`c85fcb20 00007ff9`47995d44 : 43978309`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!sk\_abort\_no\_print+0x15 [C:\b\c\b\win64\_clang\src\skia\ext\SkMemory\_new\_handler.cpp @ 41]

000000bc`c85fcb50 00007ff9`4a111464 : 0000027a`b3f31650 00007ff9`48f5e14f 00007ff9`4b432658 00007ff9`4b432658 : chrome\_child!GrOpFlushState::draw+0x1b4 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpFlushState.cpp @ 120]

000000bc`c85fcbe0 00007ff9`47990cf7 : 0000027a`be6fdca8 0000027a`be6fdca8 00000000`00000005 000000bc`c85fd540 : chrome\_child!`anonymous namespace'::AAConvexPathOp::onPrepareDraws+0x2754 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\ops\GrAAConvexPathRenderer.cpp @ 943]

000000bc`c85fd110 00007ff9`47990b41 : 0000027a`c9406730 000000bc`c85fd4b0 000000bc`c85fd304 000000bc`c85fd4b0 : chrome\_child!GrRenderTargetOpList::onPrepare+0x197 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrRenderTargetOpList.cpp @ 105]

000000bc`c85fd1b0 00007ff9`47990830 : 0000027a`bac81dd0 0000027a`bac81e18 00000000`000000ff 00000000`00000006 : chrome\_child!GrOpList::prepare+0x61 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpList.cpp @ 88]

000000bc`c85fd210 00007ff9`4798f4cf : 00000000`00000000 00000000`00000000 00000000`00000000 00005c10`68089d0f : chrome\_child!GrDrawingManager::executeOpLists+0xb0 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrDrawingManager.cpp @ 315]

000000bc`c85fd280 00007ff9`47bda0d7 : 00007ff9`4b299e07 00005c10`6808981f 00007ff9`4b299e07 00007ff9`4b299dfe : chrome\_child!GrDrawingManager::internalFlush+0x75f [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrDrawingManager.cpp @ 232]

000000bc`c85fd7a0 00007ff9`48d896b4 : 000000bc`c85fd830 0000027a`8127fd18 0000027a`c8b5fef0 00007ff9`4b3a2d30 : chrome\_child!GrContext::flush+0x27 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrContext.cpp @ 366]

1 Id: 1704.215c Suspend: 0 Teb: 000000bc`c83e4000 Unfrozen

1. **分析汇编代码**

因为源代码行号对应不上，我们看一下汇编代码：

0:000> u chrome\_child!GrOpFlushState::draw+0x1b4

chrome\_child!GrOpFlushState::draw+0x1b4 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpFlushState.cpp @ 120]:

00007ff9`47995d44 897d1c mov dword ptr [rbp+1Ch],edi

00007ff9`47995d47 897b2c mov dword ptr [rbx+2Ch],edi

00007ff9`47995d4a 48837c244000 cmp qword ptr [rsp+40h],0

00007ff9`47995d50 7507 jne chrome\_child!GrOpFlushState::draw+0x1c9 (00007ff9`47995d59)

往前一点：

0:000> u chrome\_child!GrOpFlushState::draw+0x164 L20

chrome\_child!GrOpFlushState::draw+0x164 [C:\b\c\b\win64\_clang\src\third\_party\skia\src\gpu\GrOpFlushState.cpp @ 120]:

00007ff9`47995cf4 1c85 sbb al,85h

00007ff9`47995cf6 ff754e push qword ptr [rbp+4Eh]

00007ff9`47995cf9 bf01000000 mov edi,1

00007ff9`47995cfe f00fc13d02675204 lock xadd dword ptr [chrome\_child!gCurrOpUniqueID (00007ff9`4bebc408)],edi

00007ff9`47995d06 83c701 add edi,1

00007ff9`47995d09 7539 jne chrome\_child!GrOpFlushState::draw+0x1b4 (00007ff9`47995d44)

00007ff9`47995d0b 488d057e2bad03 lea rax,[chrome\_child!`string' (00007ff9`4b468890)]

00007ff9`47995d12 4889442430 mov qword ptr [rsp+30h],rax

00007ff9`47995d17 488d0dc22bad03 lea rcx,[chrome\_child!`string' (00007ff9`4b4688e0)]

00007ff9`47995d1e 48894c2420 mov qword ptr [rsp+20h],rcx

00007ff9`47995d23 c744242817010000 mov dword ptr [rsp+28h],117h

00007ff9`47995d2b 4c8d0d7fa97b03 lea r9,[chrome\_child!`string' (00007ff9`4b1506b1)]

00007ff9`47995d32 ba17010000 mov edx,117h

00007ff9`47995d37 4531c0 xor r8d,r8d

00007ff9`47995d3a e8016a3901 call chrome\_child!SkDebugf\_FileLine (00007ff9`48d2c740)

00007ff9`47995d3f e86c683901 call chrome\_child!sk\_abort\_no\_print (00007ff9`48d2c5b0) //堆栈位置

00007ff9`47995d44 897d1c mov dword ptr [rbp+1Ch],edi

00007ff9`47995d47 897b2c mov dword ptr [rbx+2Ch],edi

00007ff9`47995d4a 48837c244000 cmp qword ptr [rsp+40h],0

00007ff9`47995d50 7507 jne chrome\_child!GrOpFlushState::draw+0x1c9 (00007ff9`47995d59)

00007ff9`47995d52 4d89a618010000 mov qword ptr [r14+118h],r12

打印一下几个关键的string

0:000> da 00007ff9`4b1506b1

00007ff9`4b1506b1 "%s(%d): fatal error: "%s"."

0:000> da 00007ff9`4b4688e0

00007ff9`4b4688e0 "../../third\_party/skia/src/gpu/o"

00007ff9`4b468900 "ps/GrOp.h"

0:000> da 00007ff9`4b468890

00007ff9`4b468890 "This should never wrap as it sho"

00007ff9`4b4688b0 "uld only be called once for each"

00007ff9`4b4688d0 " GrOp subclass."

1. **定位源码**

通过这些string和汇编，我们定位到源码调用栈为：

<https://chromium.googlesource.com/skia/+/master/src/gpu/GrOpFlushState.cpp>

void GrOpFlushState::draw(sk\_sp<const GrGeometryProcessor> gp, const GrPipeline\* pipeline,

const GrPipeline::FixedDynamicState\* fixedDynamicState,

const GrPipeline::DynamicStateArrays\* dynamicStateArrays,

const GrMesh meshes[], int meshCnt) {

SkASSERT(fOpArgs);

SkASSERT(fOpArgs->fOp);

bool firstDraw = fDraws.begin() == fDraws.end();

auto& draw = fDraws.append(&fArena);

GrDeferredUploadToken token = fTokenTracker->issueDrawToken();

if (fixedDynamicState && fixedDynamicState->fPrimitiveProcessorTextures) {

for (int i = 0; i < gp->numTextureSamplers(); ++i) {

fixedDynamicState->fPrimitiveProcessorTextures[i]->addPendingRead();

}

}

if (dynamicStateArrays && dynamicStateArrays->fPrimitiveProcessorTextures) {

int n = gp->numTextureSamplers() \* meshCnt;

for (int i = 0; i < n; ++i) {

dynamicStateArrays->fPrimitiveProcessorTextures[i]->addPendingRead();

}

}

draw.fGeometryProcessor = std::move(gp);

draw.fPipeline = pipeline;

draw.fFixedDynamicState = fixedDynamicState;

draw.fDynamicStateArrays = dynamicStateArrays;

draw.fMeshes = meshes;

draw.fMeshCnt = meshCnt;

draw.fOpID = fOpArgs->fOp->uniqueID();

if (firstDraw) {

fBaseDrawToken = token;

}

}

<https://chromium.googlesource.com/skia/+/master/src/gpu/ops/GrOp.h>

// We lazily initialize the uniqueID because currently the only user is GrAuditTrail

uint32\_t uniqueID() const {

if (kIllegalOpID == fUniqueID) {

fUniqueID = GenOpID();

}

return fUniqueID;

}

<https://chromium.googlesource.com/skia/+/master/src/gpu/ops/GrOp.h>

static uint32\_t GenOpID() { return GenID(&gCurrOpUniqueID); }

<https://chromium.googlesource.com/skia/+/master/src/gpu/ops/GrOp.h>

static uint32\_t GenID(int32\_t\* idCounter) {

// The atomic inc returns the old value not the incremented value. So we add

// 1 to the returned value.

uint32\_t id = static\_cast<uint32\_t>(sk\_atomic\_inc(idCounter)) + 1;

if (!id) {

SK\_ABORT("This should never wrap as it should only be called once for each GrOp "

"subclass.");

}

return id;

}

<https://github.com/google/skia/blob/81abc43e6f0b1a789e1bf116820c8ede68d778ab/include/core/SkPostConfig.h>

#if !defined(SkNO\_RETURN\_HINT)

# if SK\_HAS\_COMPILER\_FEATURE(attribute\_analyzer\_noreturn)

static inline void SkNO\_RETURN\_HINT() \_\_attribute\_\_((analyzer\_noreturn));

static inline void SkNO\_RETURN\_HINT() {}

# else

# define SkNO\_RETURN\_HINT() do {} while (false)

# endif

#endif

#if defined(SK\_BUILD\_FOR\_GOOGLE3)

void SkDebugfForDumpStackTrace(const char\* data, void\* unused);

void DumpStackTrace(int skip\_count, void w(const char\*, void\*), void\* arg);

# define SK\_DUMP\_GOOGLE3\_STACK() DumpStackTrace(0, SkDebugfForDumpStackTrace, nullptr)

#else

# define SK\_DUMP\_GOOGLE3\_STACK()

#endif

#ifdef SK\_BUILD\_FOR\_WIN

// permits visual studio to follow error back to source

#define SK\_DUMP\_LINE\_FORMAT(message) \

SkDebugf("%s(%d): fatal error: \"%s\"\n", \_\_FILE\_\_, \_\_LINE\_\_, message)

#else

#define SK\_DUMP\_LINE\_FORMAT(message) \

SkDebugf("%s:%d: fatal error: \"%s\"\n", \_\_FILE\_\_, \_\_LINE\_\_, message)

#endif

#ifndef SK\_ABORT

# define SK\_ABORT(message) \

do { \

SkNO\_RETURN\_HINT(); \

SK\_DUMP\_LINE\_FORMAT(message); \

SK\_DUMP\_GOOGLE3\_STACK(); \

sk\_abort\_no\_print(); \

} while (false)

#endif

这个宏展开后，就是

SkDebugf("%s(%d): fatal error: \"%s\"\n", \_\_FILE\_\_, \_\_LINE\_\_, message)

sk\_abort\_no\_print();

和汇编能对应上。

1. **分析Crash原因**

看看idCounter的初始值：

enum {

kIllegalOpID = 0,

};

int32\_t GrOp::gCurrOpUniqueID = GrOp::kIllegalOpID;

初始为0。

现在分析为何会abrt：

if (!id) {

SK\_ABORT("This should never wrap as it should only be called once for each GrOp "

"subclass.");

}

要abrt，id必须为0，表示：

uint32\_t id = static\_cast<uint32\_t>(sk\_atomic\_inc(idCounter)) + 1;

sk\_atomic\_inc(idCounter)返回的值必须是(uint32\_t)0xFFFFFFFF

排除sk\_atomic\_inc的BUG：

<https://chromium.googlesource.com/chromium/chromium/+/master/skia/ext/SkThread_chrome.cc>

int32\_t sk\_atomic\_inc(int32\_t\* addr) {

// sk\_atomic\_inc is expected to return the old value,

// Barrier\_AtomicIncrement returns the new value.

return base::subtle::NoBarrier\_AtomicIncrement(addr, 1) - 1;

}

<https://github.com/adobe/chromium/blob/cfe5bf0b51b1f6b9fe239c2a3c2f2364da9967d7/third_party/tcmalloc/chromium/src/base/atomicops-internals-x86.h>

inline Atomic64 NoBarrier\_AtomicIncrement(volatile Atomic64\* ptr,

Atomic64 increment) {

Atomic64 temp = increment;

\_\_asm\_\_ \_\_volatile\_\_("lock; xaddq %0,%1"

: "+r" (temp), "+m" (\*ptr)

: : "memory");

// temp now contains the previous value of \*ptr

return temp + increment;

}

对应汇编：

00007ff9`47995cfe f00fc13d02675204 lock xadd dword ptr [chrome\_child!gCurrOpUniqueID (00007ff9`4bebc408)],edi

00007ff9`47995d06 83c701 add edi,1

ak\_atomic\_inc比较就是一个xadd操作，没有问题。

那要让外部溢出，这个add操作应该返回 0xFFFFFFFF，4294967295为42亿次调用，便会warp。

## 验证问题

既然怀疑是此处有问题，那么我们修改内存验证来重现一下，首先，需要找一台带有显卡的物理机（之前在虚拟机里面测试，由于虚拟机没有显卡，不会跑到此GPU渲染逻辑），然后保证符号一定能加载上，直接修改内存地址：

0:018> ed 00000000`570cb120 0xFFFFFFF0

0:018> dd gCurrOpUniqueID

00000000`570cb120 fffffff0 00000002 8000021f 00000000

如上，将gCurrOpUniqueID值修改为0xFFFFFFF0，再运行，马上就Crash：

0:018> g

(211c.c68): Access violation - code c0000005 (first chance) 马上就Crash了

First chance exceptions are reported before any exception handling.

This exception may be expected and handled.

chrome\_child!base::win::`anonymous namespace'::ForceCrashOnSigAbort:

544f9fb0 c7050000000037130000 mov dword ptr ds:[0],1337h ds:002b:00000000=????????

此验证说明，前面的堆栈分析和实际是相符合的。

现在，我们测试一下，到底多久会溢出，通过秒表跑了30秒，获取前后gCurrOpUniqueID的值：

0:018> dd gCurrOpUniqueID

00000000`570cb120 0003bb6f 00000002 8000021f 00000000

0:018> g

(211c.918): Break instruction exception - code 80000003 (first chance)

ntdll!DbgBreakPoint:

00000000`77a0b1f0 cc int 3

0:018> dd gCurrOpUniqueID

00000000`570cb120 0003f7ff 00000002 8000021f 00000000

计算一下，30秒增长53021，一天53021\*2\*60\*24增长 152700480，需要28天左右才会rewind。考虑到调试时候速度慢一些，假设慢4倍，那正好一周左右会溢出。

溢出后新产生的Core dump，会出现多个版本，其中一个是和客户的Core dump完全能对应上的，另一个和之前我们自己重现出来的堆栈破坏Core dump也能对应上，此堆栈如下：

FAULTING\_IP:

chrome\_child!base::win::IsDeviceUsedAsATablet+a0 [C:\b\c\b\win\_clang\src\base\win\win\_util.cc @ 516]

544f9fb0 c7050000000037130000 mov dword ptr ds:[0],1337h

EXCEPTION\_RECORD: ffffffff -- (.exr 0xffffffffffffffff)

ExceptionAddress: 544f9fb0 (chrome\_child!base::win::IsDeviceUsedAsATablet+0x000000a0)

ExceptionCode: c0000005 (Access violation)

ExceptionFlags: 00000000

NumberParameters: 2

Parameter[0]: 00000001

Parameter[1]: 00000000

Attempt to write to address 00000000

PROCESS\_NAME: chrome.exe

ERROR\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_PARAMETER1: 00000001

EXCEPTION\_PARAMETER2: 00000000

WRITE\_ADDRESS: 00000000

FOLLOWUP\_IP:

chrome\_child!base::win::IsDeviceUsedAsATablet+a0 [C:\b\c\b\win\_clang\src\base\win\win\_util.cc @ 516]

544f9fb0 c7050000000037130000 mov dword ptr ds:[0],1337h

MOD\_LIST: <ANALYSIS/>

CHKIMG\_EXTENSION: !chkimg -lo 50 -d !chrome\_child

5309526a - chrome\_child!base::trace\_event::MemoryDumpProviderInfo::MemoryDumpProviderInfo+5a

[ 12:02 ]

530952cd-530952ce 2 bytes - chrome\_child!base::internal::LockImpl::Lock+4d (+0x63)

[ c6 3a:46 3b ]

53095331 - chrome\_child!base::debug::ScopedLockAcquireActivity::ScopedLockAcquireActivity+41 (+0x64)

[ 62:e2 ]

53095383-53095384 2 bytes - chrome\_child!base::debug::GlobalActivityTracker::ScopedThreadActivity::ScopedThreadActivity+43 (+0x52)

[ d9 d4:69 d5 ]

544f9f30-544f9f61 50 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+20 (+0x1464bad)

[ 85 f6 0f 84 33 03 00 00:8a 45 ec 89 f1 88 44 24 ]

544f9f63-544f9f8f 45 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+53 (+0x33)

[ 04 03 00 00 c7 04 24 5f:e8 af ee 02 02 83 c4 08 ]

544f9f91-544f9fec 92 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+81 (+0x2e)

[ 00 00 85 f6 0f 84 d0 02:31 db 8b 4d f0 31 e9 e8 ]

544f9fee-544f9ffb 14 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+de (+0x5d)

[ f9 1f 0f 83 de 01 00 00:e1 00 00 00 8a 45 d8 89 ]

544f9ffe-544fa014 23 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+ee (+0x10)

[ 04 c7 44 24 0c 1f 00 00:08 91 0d 6e 56 c7 04 24 ]

544fa016-544fa030 27 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+106 (+0x18)

[ e8 53 a9 ba fe 83 ec 10:00 c7 04 24 5f 00 00 00 ]

544fa032-544fa056 37 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+122 (+0x1c)

[ 00 00 c7 44 24 04 91 0d:ff 15 28 6e fa 56 83 ec ]

544fa058-544fa0a0 73 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+148 (+0x26)

[ 72 02 8b 0e c7 44 01 0c:0f 83 9f 00 00 00 8a 45 ]

544fa0a2-544fa0f1 80 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+192 (+0x4a)

[ ec 08 85 c0 74 20 8d 4d:de 01 00 00 8a 45 dc 89 ]

544fa0f3-544fa12f 61 bytes - chrome\_child!base::win::IsDeviceUsedAsATablet+1e3 (+0x51)

[ 85 c0 0f 84 83 00 00 00:02 c6 47 36 00 e9 1e 02 ]

508 errors : !chrome\_child (5309526a-544fa12f)

FAULTING\_THREAD: 000031c4

BUGCHECK\_STR: APPLICATION\_FAULT\_MEMORY\_CORRUPTION\_NULL\_POINTER\_WRITE\_LARGE

PRIMARY\_PROBLEM\_CLASS: MEMORY\_CORRUPTION\_LARGE

DEFAULT\_BUCKET\_ID: MEMORY\_CORRUPTION\_LARGE

LAST\_CONTROL\_TRANSFER: from 5655b868 to 544f9fb0

STACK\_TEXT:

0036d52c 5655b868 00000016 5465f605 0036d564 chrome\_child!base::win::IsDeviceUsedAsATablet+0xa0 [C:\b\c\b\win\_clang\src\base\win\win\_util.cc @ 516]

0036d53c 53607be6 00000000 00000000 05a30998 chrome\_child!\_get\_fmode+0x2b [minkernel\crts\ucrt\src\appcrt\lowio\setmode.cpp @ 121]

0036d564 53607a31 00000000 0986006c 00000000 chrome\_child!GrOpFlushState::draw+0x166 [C:\b\c\b\win\_clang\src\third\_party\skia\src\gpu\GrOpFlushState.cpp @ 120]

0036d66c 5334fa98 0036e7b0 00008000 ac34bc51 chrome\_child!GrMeshDrawOp::Target::draw+0x11 [C:\b\c\b\win\_clang\src\third\_party\skia\src\gpu\ops\GrMeshDrawOp.h @ 93]

0036e530 530911da 05a30998 0997b560 00000000 chrome\_child!sk\_malloc\_flags+0x68 [C:\b\c\b\win\_clang\src\skia\ext\SkMemory\_new\_handler.cpp @ 117]

0036e534 05a30998 0997b560 00000000 00000000 chrome\_child!base::allocator::WinHeapFree+0x1a [C:\b\c\b\win\_clang\src\base\allocator\winheap\_stubs\_win.cc @ 43]

WARNING: Frame IP not in any known module. Following frames may be wrong.

0036e538 0997b560 00000000 00000000 00000000 0x5a30998

0036e53c 00000000 00000000 00000000 00000000 0x997b560

STACK\_COMMAND: ~0s; .ecxr ; kb

SYMBOL\_NAME: memory\_corruption!chrome\_child

FOLLOWUP\_NAME: MachineOwner

MODULE\_NAME: memory\_corruption

IMAGE\_NAME: memory\_corruption

DEBUG\_FLR\_IMAGE\_TIMESTAMP: 0

FAILURE\_BUCKET\_ID: MEMORY\_CORRUPTION\_LARGE\_c0000005\_memory\_corruption!chrome\_child

BUCKET\_ID: APPLICATION\_FAULT\_MEMORY\_CORRUPTION\_NULL\_POINTER\_WRITE\_LARGE\_memory\_corruption!chrome\_child

WATSON\_STAGEONE\_URL: http://watson.microsoft.com/StageOne/chrome\_exe/70\_0\_3538\_102/5be3c2d0/chrome\_child\_dll/70\_0\_3538\_102/5be3c2d0/c0000005/01469fb0.htm?Retriage=1

Followup: MachineOwner

---------

此堆栈破坏版本，没法跟到根因，导致之前一次分析自己的Core dump时，误以为是踩内存导致，其实应该是Crash handler没有dump好内存。

还有一种堆栈，是GPU Watch dog超时（15秒），这个Crash的是GPU进程，之前的是render进程：

Executable search path is:

Windows 7 Version 7601 (23391) MP (2 procs) Free x64

Product: WinNt

Machine Name:

Debug session time: Sat Nov 10 12:11:41.000 2018 (GMT+8) //这个时间，应该是客户的老机器

System Uptime: not available

Process Uptime: 0 days 13:31:03.000

................................................................

........................

Loading unloaded module list

0:001> !analyze -v

Failed calling InternetOpenUrl, GLE=12002

FAULTING\_IP:

chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+1c1 [C:\b\c\b\win64\_clang\src\gpu\ipc\service\gpu\_watchdog\_thread.cc @ 505]

000007fe`ebec081f c704250000000037130000 mov dword ptr [0],1337h

EXCEPTION\_RECORD: ffffffffffffffff -- (.exr 0xffffffffffffffff)

ExceptionAddress: 000007feebec081f (chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+0x00000000000001c1)

ExceptionCode: c0000005 (Access violation)

ExceptionFlags: 00000000

NumberParameters: 2

Parameter[0]: 0000000000000001

Parameter[1]: 0000000000000000

Attempt to write to address 0000000000000000

DEFAULT\_BUCKET\_ID: NULL\_POINTER\_WRITE

PROCESS\_NAME: chrome.exe

ERROR\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_CODE: (NTSTATUS) 0xc0000005 - 0x%08lx

EXCEPTION\_PARAMETER1: 0000000000000001

EXCEPTION\_PARAMETER2: 0000000000000000

WRITE\_ADDRESS: 0000000000000000

FOLLOWUP\_IP:

chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+1c1 [C:\b\c\b\win64\_clang\src\gpu\ipc\service\gpu\_watchdog\_thread.cc @ 505]

000007fe`ebec081f c704250000000037130000 mov dword ptr [0],1337h

MOD\_LIST: <ANALYSIS/>

NTGLOBALFLAG: 0

FAULTING\_THREAD: 000000000000135c

PRIMARY\_PROBLEM\_CLASS: NULL\_POINTER\_WRITE

BUGCHECK\_STR: APPLICATION\_FAULT\_NULL\_POINTER\_WRITE

LAST\_CONTROL\_TRANSFER: from 000007feebec0489 to 000007feebec081f

STACK\_TEXT:

00000000`0349f440 000007fe`ebec0489 : 00000000`00000001 00000000`00000000 0000deed`2785490f 000000d9`e819a530 : chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+0x1c1 [C:\b\c\b\win64\_clang\src\gpu\ipc\service\gpu\_watchdog\_thread.cc @ 505]

00000000`0349f520 000007fe`ea16d2cc : 00000000`00000000 00000000`7751dba8 00000000`00000000 000007fe`00000002 : chrome\_child!gpu::GpuWatchdogThread::OnCheckTimeout+0x61 [C:\b\c\b\win64\_clang\src\gpu\ipc\service\gpu\_watchdog\_thread.cc @ 368]

00000000`0349f5d0 000007fe`ea16cb37 : 00000000`00000000 00000000`028674d0 00000000`00000001 00000000`0349f758 : chrome\_child!base::debug::TaskAnnotator::RunTask+0x12c [C:\b\c\b\win64\_clang\src\base\debug\task\_annotator.cc @ 101]

00000000`0349f6f0 000007fe`ea1682bd : 00000000`00000048 00000000`00000001 00000000`00000000 000007fe`ee92d8d0 : chrome\_child!base::MessageLoop::RunTask+0x247 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 423]

00000000`0349f850 000007fe`ea1678bc : 00000000`00000000 000007fe`ea14bb94 00000000`0349fa70 000007fe`ee92d6d8 : chrome\_child!base::MessageLoop::DoDelayedWork+0x18d [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 521]

00000000`0349f9c0 000007fe`ea167621 : 00000000`02867250 00000000`00000000 00000000`02867260 00000000`02867250 : chrome\_child!base::MessagePumpDefault::Run+0x4c [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_pump\_default.cc @ 42]

00000000`0349fa20 000007fe`ea163f90 : 00000000`00000000 00000000`00000000 000007fe`ea13d4e2 00000000`00000000 : chrome\_child!base::RunLoop::Run+0x31 [C:\b\c\b\win64\_clang\src\base\run\_loop.cc @ 108]

00000000`0349fa50 000007fe`eb8670b4 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!base::Thread::ThreadMain+0x180 [C:\b\c\b\win64\_clang\src\base\threading\thread.cc @ 340]

00000000`0349fae0 00000000`773c59bd : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!base::`anonymous namespace'::ThreadFunc+0xf4 [C:\b\c\b\win64\_clang\src\base\threading\platform\_thread\_win.cc @ 94]

00000000`0349fb60 00000000`774fa2e1 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : kernel32!BaseThreadInitThunk+0xd

00000000`0349fb90 00000000`00000000 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : ntdll!RtlUserThreadStart+0x1d

STACK\_COMMAND: ~1s; .ecxr ; kb

SYMBOL\_STACK\_INDEX: 0

SYMBOL\_NAME: chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+1c1

FOLLOWUP\_NAME: MachineOwner

MODULE\_NAME: chrome\_child

IMAGE\_NAME: chrome\_child.dll

DEBUG\_FLR\_IMAGE\_TIMESTAMP: 5b96fc9f

FAILURE\_BUCKET\_ID: NULL\_POINTER\_WRITE\_c0000005\_chrome\_child.dll!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang

BUCKET\_ID: X64\_APPLICATION\_FAULT\_NULL\_POINTER\_WRITE\_chrome\_child!gpu::GpuWatchdogThread::DeliberatelyTerminateToRecoverFromHang+1c1

WATSON\_STAGEONE\_URL: http://watson.microsoft.com/StageOne/chrome\_exe/69\_0\_3497\_92/5b96fd04/chrome\_child\_dll/69\_0\_3497\_92/5b96fc9f/c0000005/01da081f.htm?Retriage=1

Followup: MachineOwner

此进程是GPU渲染进程：

0:001> !peb

PEB at 000007fffffdf000

CommandLine '"C:\Users\Administrator\AppData\Local\Google\Chrome\Application\chrome.exe" --type=gpu-process --field-trial-handle=1028,15351644294314642440,598078252489282679,131072 --gpu-preferences=KAAAAAAAAACAAwBAAQAAAAAAAAAAAGAAAAAAAAAAAAAIAAAAAAAAACgAAAAEAAAAIAAAAAAAAAAoAAAAAAAAADAAAAAAAAAAOAAAAAAAAAAQAAAAAAAAAAAAAAAKAAAAEAAAAAAAAAAAAAAACwAAABAAAAAAAAAAAQAAAAoAAAAQAAAAAAAAAAEAAAALAAAA --service-request-channel-token=2922249399616643618 --mojo-platform-channel-handle=2076 --ignored=" --type=renderer " /prefetch:2'

搜索到类似BUG <https://bugs.chromium.org/p/chromium/issues/detail?id=597700> 看讨论说升级intel驱动可解决。

下断点：

bp chrome\_child!gpu::GpuWatchdogThread::OnCheckTimeout

bp chrome\_child!gpu::GpuWatchdogThread::OnCheck

0:001:x86> bl

0 e x86 10a48680 0001 (0001) 0:\*\*\*\* chrome\_child!gpu::GpuWatchdogThread::OnCheckTimeout

2 e x86 10a4817c 0001 (0001) 0:\*\*\*\* chrome\_child!gpu::GpuWatchdogThread::OnCheck

0:001:x86> g

Breakpoint 2 hit

chrome\_child!gpu::GpuWatchdogThread::OnCheck: //在GPU硬件加速开启的情况下定时15秒会断下来，不开启的时候，断不下来。

10a4817c 55 push ebp

0:001:x86> kv

ChildEBP RetAddr Args to Child

038af968 10338e22 00000000 10a4817c 00000000 chrome\_child!gpu::GpuWatchdogThread::OnCheck (CONV: thiscall) [C:\b\c\b\win\_clang\src\gpu\ipc\service\gpu\_watchdog\_thread.cc @ 292]

038af984 0f07c562 04e51808 02440000 00000000 chrome\_child!base::internal::Invoker<base::internal::BindState<void (blink::scheduler::MainThreadSchedulerImpl::\*)(bool) \_\_attribute\_\_((thiscall)),base::WeakPtr<blink::scheduler::MainThreadSchedulerImpl>,bool>,void ()>::RunOnce+0x44 (CONV: cdecl) [C:\b\c\b\win\_clang\src\base\bind\_internal.h @ 662]

038af9f0 0f07bbd3 1267f4c0 038afa98 038afa80 chrome\_child!base::debug::TaskAnnotator::RunTask+0xe2 (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\debug\task\_annotator.cc @ 101]

038afa70 0f07baf3 038afa98 0f074740 f687dac3 chrome\_child!base::MessageLoop::RunTask+0xb3 (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\message\_loop\message\_loop.cc @ 436]

038afa90 0f0752a4 00000000 12848700 128486ac chrome\_child!base::MessageLoop::DeferOrRunPendingTask+0x53 (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\message\_loop\message\_loop.cc @ 445]

038afb58 0f074b0a 02465a70 02465a70 02465a68 chrome\_child!base::MessageLoop::DoDelayedWork+0x134 (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\message\_loop\message\_loop.cc @ 558]

038afb78 0f074aaf 0247b6c8 038afbb8 038afb98 chrome\_child!base::MessagePumpDefault::Run+0x3a (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\message\_loop\message\_pump\_default.cc @ 42]

038afb88 0f0748fe 00000001 0247b5e0 038afba0 chrome\_child!base::MessageLoop::Run+0x1f (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\message\_loop\message\_loop.cc @ 386]

038afb98 0f0748cb 038afbe0 0f073105 038afbb8 chrome\_child!base::RunLoop::Run+0x2e (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\run\_loop.cc @ 108]

038afba0 0f073105 038afbb8 0247b6c8 00000000 chrome\_child!base::Thread::Run+0xb (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\threading\thread.cc @ 263]

038afbe0 1047f865 0247b5e0 000001c8 000001c8 chrome\_child!base::Thread::ThreadMain+0x165 (CONV: thiscall) [C:\b\c\b\win\_clang\src\base\threading\thread.cc @ 360]

038afc04 7721343d 0246d408 038afc50 77b99832 chrome\_child!base::`anonymous namespace'::ThreadFunc+0x95 (CONV: stdcall) [C:\b\c\b\win\_clang\src\base\threading\platform\_thread\_win.cc @ 103]

038afc10 77b99832 0246d408 a7488ffb 00000000 KERNEL32!BaseThreadInitThunk+0xe (FPO: [Non-Fpo])

038afc50 77b99805 1047f7d0 0246d408 ffffffff ntdll\_77b60000!\_\_RtlUserThreadStart+0x70 (FPO: [Non-Fpo])

038afc68 00000000 1047f7d0 0246d408 00000000 ntdll\_77b60000!\_RtlUserThreadStart+0x1b (FPO: [Non-Fpo])

还有一种是内存耗光的，但是比较早了：

Windows 7 Version 7601 (23391) MP (2 procs) Free x64

Product: WinNt

Machine Name:

Debug session time: Sun Nov 18 06:58:52.000 2018 (UTC + 8:00)

System Uptime: not available

Process Uptime: 4 days 13:20:33.000

版本还是69.0.3497.92

0:009> !analyze -v

\*\*\* WARNING: Unable to verify checksum for KERNELBASE.dll

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* Exception Analysis \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* WARNING: Unable to verify checksum for chrome\_child.dll

\*\*\* WARNING: Unable to verify checksum for kernel32.dll

\*\*\* WARNING: Unable to verify checksum for USER32.dll

\*\*\* WARNING: Unable to verify checksum for ole32.dll

\*\*\* WARNING: Unable to verify checksum for chrome\_elf.dll

Unable to load image C:\Program Files (x86)\360\360Safe\safemon\safemon64.dll, Win32 error 0n2

\*\*\* WARNING: Unable to verify timestamp for safemon64.dll

\*\*\* ERROR: Module load completed but symbols could not be loaded for safemon64.dll

Failed calling InternetOpenUrl, GLE=12002

FAULTING\_IP:

KERNELBASE!RaiseException+39

000007fe`fd12a06d 4881c4c8000000 add rsp,0C8h

EXCEPTION\_RECORD: ffffffffffffffff -- (.exr 0xffffffffffffffff)

ExceptionAddress: 000007fefd12a06d (KERNELBASE!RaiseException+0x0000000000000039)

ExceptionCode: e0000008

ExceptionFlags: 00000001

NumberParameters: 1

Parameter[0]: 0000000045c5c040

DEFAULT\_BUCKET\_ID: APPLICATION\_FAULT

PROCESS\_NAME: chrome.exe

ERROR\_CODE: (NTSTATUS) 0xe0000008 - <Unable to get error code text>

EXCEPTION\_CODE: (NTSTATUS) 0xe0000008 - <Unable to get error code text>

EXCEPTION\_PARAMETER1: 0000000045c5c040

MOD\_LIST: <ANALYSIS/>

NTGLOBALFLAG: 0

FAULTING\_THREAD: 0000000000001798

PRIMARY\_PROBLEM\_CLASS: APPLICATION\_FAULT

BUGCHECK\_STR: APPLICATION\_FAULT\_APPLICATION\_FAULT

LAST\_CONTROL\_TRANSFER: from 000007feeb854284 to 000007fefd12a06d

STACK\_TEXT:

00000000`04aae590 000007fe`eb854284 : 000007fe`eb854260 00000000`00000000 00000000`00000000 00000000`012a8ed0 : KERNELBASE!RaiseException+0x39

00000000`04aae660 000007fe`ebc26767 : 00000000`00000000 00000000`00000000 000007fe`edf9ff98 00000000`45c5c040 : chrome\_child!base::`anonymous namespace'::OnNoMemory+0x24 [C:\b\c\b\win64\_clang\src\base\process\memory\_win.cc @ 55]

00000000`04aae690 000007fe`ea121136 : 00000000`00000003 00000000`007f957c 00000000`775e03c8 000007fe`ea49b5ad : chrome\_child!base::allocator::WinCallNewHandler+0x17 [C:\b\c\b\win64\_clang\src\base\allocator\winheap\_stubs\_win.cc @ 90]

00000000`04aae6c0 000007fe`ea181b98 : 00000000`01302f50 00000000`04934395 00000000`00000000 000007fe`ea125d27 : chrome\_child!malloc+0x46 [C:\b\c\b\win64\_clang\src\base\allocator\allocator\_shim\_override\_ucrt\_symbols\_win.h @ 53]

00000000`04aae700 000007fe`ea181a1f : 00000000`01302e50 00000000`01302e40 00000000`04aae880 00000000`04aae930 : chrome\_child!base::circular\_deque<base::sequence\_manager::internal::TaskQueueImpl::Task>::ExpandCapacityIfNecessary+0x88 [C:\b\c\b\win64\_clang\src\base\containers\circular\_deque.h @ 960]

00000000`04aae780 000007fe`ea1818c2 : 000007fe`edf9ff98 00000000`00000048 00000000`3a4ab518 000007fe`ea12111b : chrome\_child!base::sequence\_manager::internal::TaskQueueImpl::PushOntoImmediateIncomingQueueLocked+0x6f [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\task\_queue\_impl.cc @ 311]

00000000`04aae800 000007fe`ea17753c : 00000000`04aaebb8 000007fe`ea125d83 00000000`0373c740 000007fe`ea499461 : chrome\_child!base::sequence\_manager::internal::TaskQueueImpl::PostImmediateTaskImpl+0xd2 [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\task\_queue\_impl.cc @ 209]

00000000`04aae9f0 000007fe`ea177458 : 00000000`012bb5c0 00000000`00000000 00000000`00000000 00000000`04aaeb18 : chrome\_child!base::sequence\_manager::internal::TaskQueueImpl::PostDelayedTask+0x6c [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\task\_queue\_impl.cc @ 192]

00000000`04aaeab0 000007fe`ea177316 : 00000000`012bb740 000007fe`ea5ad0a9 00000000`00000001 0000123a`865fd636 : chrome\_child!base::sequence\_manager::TaskQueue::PostTaskWithMetadata+0xa8 [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\task\_queue.cc @ 116]

00000000`04aaeb80 000007fe`ea16c7bf : 00000000`00000038 00000000`01309580 00000000`1b9d9da0 000007fe`edc8150b : chrome\_child!blink::scheduler::TaskQueueWithTaskType::PostDelayedTask+0x7e [C:\b\c\b\win64\_clang\src\third\_party\blink\renderer\platform\scheduler\child\task\_queue\_with\_task\_type.cc @ 37]

00000000`04aaec50 000007fe`ea5ace8a : 00000000`00000002 000007fe`ea5acd78 00000000`04aaedc0 000007fe`ea138c80 : chrome\_child!base::TaskRunner::PostTask+0x3f [C:\b\c\b\win64\_clang\src\base\task\_runner.cc @ 44]

00000000`04aaeca0 000007fe`ea477f1d : 00000000`00000000 00000000`04aaeea0 000007fe`edd5bc30 00000000`00000000 : chrome\_child!cc::ProxyImpl::DidReceiveCompositorFrameAckOnImplThread+0xea [C:\b\c\b\win64\_clang\src\cc\trees\proxy\_impl.cc @ 297]

00000000`04aaeda0 000007fe`ea1cf431 : 00000000`00000000 00000000`04aaedc0 00000000`04aaedc0 00000000`00000000 : chrome\_child!viz::mojom::CompositorFrameSinkClientStubDispatch::Accept+0x31b [C:\b\c\b\win64\_clang\src\out\Release\_x64\gen\services\viz\public\interfaces\compositing\compositor\_frame\_sink.mojom.cc @ 1387]

00000000`04aaee80 000007fe`ea1cef77 : 00000000`04aaeffc 000007fe`ea1ce782 000007fe`ea49ddc2 000007fe`ee8c7af0 : chrome\_child!mojo::internal::MultiplexRouter::ProcessIncomingMessage+0x17b [C:\b\c\b\win64\_clang\src\mojo\public\cpp\bindings\lib\multiplex\_router.cc @ 868]

00000000`04aaef50 000007fe`ea1ce306 : 00000000`04aaf1a8 000007fe`ea1be884 0def32d8`00020007 00000000`012bb350 : chrome\_child!mojo::internal::MultiplexRouter::Accept+0xc7 [C:\b\c\b\win64\_clang\src\mojo\public\cpp\bindings\lib\multiplex\_router.cc @ 594]

00000000`04aaf110 000007fe`ea1ce181 : 00000000`04aaf2f0 00000000`066304d0 00000000`04aaf340 000007fe`ea9eb007 : chrome\_child!mojo::Connector::ReadSingleMessage+0xfe [C:\b\c\b\win64\_clang\src\mojo\public\cpp\bindings\lib\connector.cc @ 456]

00000000`04aaf2b0 000007fe`ea1ce05f : 00000000`00000000 00000000`04ad8f88 00000000`04aaf600 000007fe`ea1cdf85 : chrome\_child!mojo::Connector::ReadAllAvailableMessages+0x63 [C:\b\c\b\win64\_clang\src\mojo\public\cpp\bindings\lib\connector.cc @ 486]

00000000`04aaf320 000007fe`ea16d2cc : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`0382b9f8 : chrome\_child!mojo::SimpleWatcher::OnHandleReady+0xab [C:\b\c\b\win64\_clang\src\mojo\public\cpp\system\simple\_watcher.cc @ 274]

00000000`04aaf390 000007fe`ea1bc986 : 00000000`04aaf618 00000000`04aaf680 00000000`04ad8f18 000007fe`ea165781 : chrome\_child!base::debug::TaskAnnotator::RunTask+0x12c [C:\b\c\b\win64\_clang\src\base\debug\task\_annotator.cc @ 101]

00000000`04aaf4b0 000007fe`ea16d2cc : 00000000`3a4ab2f0 00000000`00000001 000007fe`ea1bce04 00000000`03825a18 : chrome\_child!base::sequence\_manager::internal::ThreadControllerImpl::DoWork+0x1b6 [C:\b\c\b\win64\_clang\src\base\task\sequence\_manager\thread\_controller\_impl.cc @ 170]

00000000`04aaf6f0 000007fe`ea16cb37 : 00000000`3a4ab2f0 efefefef`efefefef 000007fe`ea1bce04 000007fe`ea1bce04 : chrome\_child!base::debug::TaskAnnotator::RunTask+0x12c [C:\b\c\b\win64\_clang\src\base\debug\task\_annotator.cc @ 101]

00000000`04aaf810 000007fe`ea167ac8 : 000007fe`edf9ca2c 000007fe`edf9c935 0000123a`865fc596 0000123a`865fc5f6 : chrome\_child!base::MessageLoop::RunTask+0x247 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 423]

00000000`04aaf970 000007fe`ea167909 : 00000000`00000000 000007fe`ea14bb94 00000000`04aafc10 000007fe`ee92d6d8 : chrome\_child!base::MessageLoop::DoWork+0x198 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_loop.cc @ 480]

00000000`04aafb60 000007fe`ea167621 : 00000000`00000000 00000000`00000000 00000000`037df9f0 00000000`037df9e0 : chrome\_child!base::MessagePumpDefault::Run+0x99 [C:\b\c\b\win64\_clang\src\base\message\_loop\message\_pump\_default.cc @ 37]

00000000`04aafbc0 000007fe`ea163f90 : 00000000`00000000 00000000`00000000 000007fe`ea13d4e2 00000000`00000000 : chrome\_child!base::RunLoop::Run+0x31 [C:\b\c\b\win64\_clang\src\base\run\_loop.cc @ 108]

00000000`04aafbf0 000007fe`eb8670b4 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!base::Thread::ThreadMain+0x180 [C:\b\c\b\win64\_clang\src\base\threading\thread.cc @ 340]

00000000`04aafc80 00000000`773c59bd : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : chrome\_child!base::`anonymous namespace'::ThreadFunc+0xf4 [C:\b\c\b\win64\_clang\src\base\threading\platform\_thread\_win.cc @ 94]

00000000`04aafd00 00000000`774fa2e1 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : kernel32!BaseThreadInitThunk+0xd

00000000`04aafd30 00000000`00000000 : 00000000`00000000 00000000`00000000 00000000`00000000 00000000`00000000 : ntdll!RtlUserThreadStart+0x1d

STACK\_COMMAND: ~9s; .ecxr ; kb

FOLLOWUP\_IP:

chrome\_child!base::`anonymous namespace'::OnNoMemory+24 [C:\b\c\b\win64\_clang\src\base\process\memory\_win.cc @ 55]

000007fe`eb854284 b9080000e0 mov ecx,0E0000008h

SYMBOL\_STACK\_INDEX: 1

SYMBOL\_NAME: chrome\_child!base::`anonymous namespace'::OnNoMemory+24

FOLLOWUP\_NAME: MachineOwner

MODULE\_NAME: chrome\_child

IMAGE\_NAME: chrome\_child.dll

DEBUG\_FLR\_IMAGE\_TIMESTAMP: 5b96fc9f

FAILURE\_BUCKET\_ID: APPLICATION\_FAULT\_e0000008\_chrome\_child.dll!base::\_anonymous\_namespace\_::OnNoMemory

BUCKET\_ID: X64\_APPLICATION\_FAULT\_APPLICATION\_FAULT\_chrome\_child!base::\_anonymous\_namespace\_::OnNoMemory+24

WATSON\_STAGEONE\_URL: http://watson.microsoft.com/StageOne/chrome\_exe/69\_0\_3497\_92/5b96fd04/KERNELBASE\_dll/6\_1\_7601\_23391/56e9ab2a/e0000008/0001a06d.htm?Retriage=1

Followup: MachineOwner

---------

## 简化重现

因为这是一个标准的2d draw调用，不需要用我们的代码，同事专门写了一段2d Canvas绘制代码，便可验证此问题（需要运行好几天），另外，测试的时候，我还发现，把Chrome浏览器最小化或不是激活Tab，都会停止绘图，需要重现得保证Chrome一直显示：

<html lang="en">

<head>

<meta charset="UTF-8">

<title>2d draw</title>

<script type="text/javascript">

function draw () {

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.beginPath();

ctx.arc(100,75,50,0,2\*Math.PI);

ctx.stroke();

}

window.onload = function () {

var time = +new Date();

var test = document.getElementById('test');

var func;

window.requestAnimationFrame(func = function () {

var newTime = +new Date();

test.innerHTML = newTime - time;

time = +new Date();

for(var i = 0; i < 100000; i++)

draw();

window.requestAnimationFrame(func);

});

}

</script>

</head>

<body>

<div id="test"></div>

<canvas id="myCanvas" width="500" height="500"></canvas>

</body>

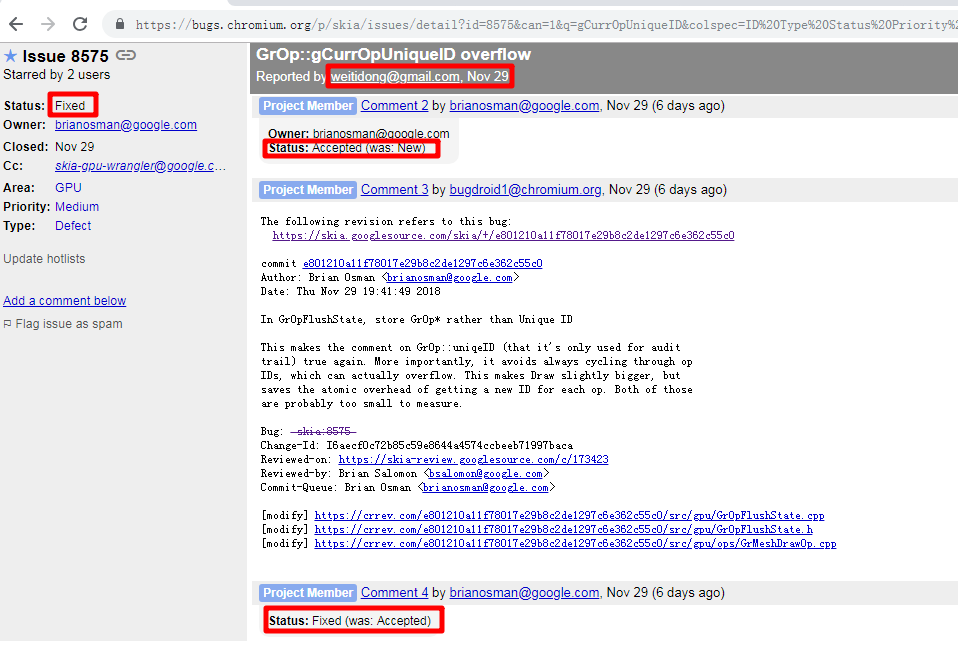
</html>

## 解决方法

目前看，最好的方法是更换为非Chrome浏览器，此问题已经反馈给Chrome官方。

<https://bugs.chromium.org/p/skia/issues/detail?id=8575&can=1&q=gCurrOpUniqueID&colspec=ID%20Type%20Status%20Priority%20M%20Area%20Owner%20Summary>

当天晚上，Chrome就修复了此问题，预计后面的新版本就不会有此问题：



2018/11/29