

authorAlex Hung <alex.hung@amd.com>2023-01-11 09:54:11 -0700

committerGreg Kroah-Hartman <gregkh@linuxfoundation.org>2023-03-22 13:37:57 +0100

commit**e12b95680821b9880cd9992c0f3555389363604f** (patch)

tree**e627331ad07ec26b01e70672fa40dea51c1059fb**

parent**55a7c647ebf6e376c45d8322568dd6eb71937139** (diff)

download**linux-e12b95680821b9880cd9992c0f3555389363604f.tar.gz**

diff options

context:

3

space:

include

mode:

unified

drm/amd/display: fix shift-out-of-bounds in CalculateVMAndRowBytes

[Upstream commit 031f196d1b1b6d5dfcb0533b431e3ab1750e6189]

[WHY]
When PTEBufferSizeInRequests is zero, UBSAN reports the following warning because dml_log2 returns an unexpected negative value:

shift exponent 4294966273 is too large for 32-bit type 'int'

[HOW]

In the case PTEBufferSizeInRequests is zero, skip the dml_log2() and assign the result directly.

Reviewed-by: Jun Lei <Jun.Lei@amd.com>
Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
Signed-off-by: Alex Hung <alex.hung@amd.com>
Tested-by: Daniel Wheeler <daniel.wheeler@amd.com>
Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

-rw-r--r--

drivers/gpu/drm/amd/display/dc/dml/dcn30/display_mode_vba_30.c

5

1 files changed, 4 insertions, 1 deletions

```
diff --git a/drivers/gpu/drm/amd/display/dc/dml/dcn30/display_mode_vba_30.c b/drivers/gpu/drm/amd/display/dc/dml/dcn30/display_mode_vba_30.c
index 379729b0284748..c3d75e56410cc7 100644
--- a/drivers/gpu/drm/amd/display/dc/dml/dcn30/display_mode_vba_30.c
+++ b/drivers/gpu/drm/amd/display/dc/dml/dcn30/display_mode_vba_30.c
@@ -1802,7 +1802,10 @@ static unsigned int CalculateVMAndRowBytes(
     }

     if (SurfaceTiling == dm_sw_linear) {
-        *dppte_row_height = dml_min(128, 1 << (unsigned int) dml_floor(dml_log2(PTEBufferSizeInRequests * *PixelPTEReqWidth / Pitch), 1));
+        if (PTEBufferSizeInRequests == 0)
+            *dppte_row_height = 1;
+        else
+            *dppte_row_height = dml_min(128, 1 << (unsigned int) dml_floor(dml_log2(PTEBufferSizeInRequests * *PixelPTEReqWidth / Pitch), 1));
         *dppte_row_width_ub = (dml_ceil(((double) SwathWidth - 1) / *PixelPTEReqWidth, 1) + 1) * *PixelPTEReqWidth;
         *PixelPTEBytesPerRow = *dppte_row_width_ub / *PixelPTEReqWidth * *PTERequestSize;
     } else if (ScanDirection != dm_vert) {
```