



author Michal Wajdeczko <michal.wajdeczko@intel.com> 2025-01-31 19:25:02 +0100
committer Greg Kroah-Hartman <gregkh@linuxfoundation.org> 2025-04-20 10:15:23 +0200
commit [2eec2fa8666dcecebae33a565a818c9de9af8b50](#) (patch)
tree [12c672529079c0799f21ddb18cbfc0408c37e225](#)
parent [4a5f14246bc484db3d06969ddd9831e6bdcf9a29](#) (diff)
download [linux-2eec2fa8666dcecebae33a565a818c9de9af8b50.tar.gz](#)

diff options

context:
space:
mode:

drm/xe/vf: Don't try to trigger a full GT reset if VF

[Upstream commit 459777724d306315070d24608fcd89aea85516d6]

VFs don't have access to the GDRST(0x941c) register that driver uses to reset a GT. Attempt to trigger a reset using debugfs:

```
$ cat /sys/kernel/debug/dri/0000:00:02.1/gt0/force_reset
```

or due to a hang condition detected by the driver leads to:

```
[ ] xe 0000:00:02.1: [drm] GT0: trying reset from force_reset [xe]
[ ] xe 0000:00:02.1: [drm] GT0: reset queued
[ ] xe 0000:00:02.1: [drm] GT0: reset started
[ ] -----[ cut here ]-----
[ ] xe 0000:00:02.1: [drm] GT0: VF is trying to write 0x1 to an inaccessible register 0x941c+0x0
[ ] WARNING: CPU: 3 PID: 3069 at drivers/gpu/drm/xe/xe_gt_sriov_vf.c:996 xe_gt_sriov_vf_write32+0xc6/0x580 [xe]
[ ] RIP: 0010:xe_gt_sriov_vf_write32+0xc6/0x580 [xe]
[ ] Call Trace:
[ ] <TASK>
[ ] ? show_regs+0x6c/0x80
[ ] ? __warn+0x93/0x1c0
[ ] ? xe_gt_sriov_vf_write32+0xc6/0x580 [xe]
[ ] ? report_bug+0x182/0x1b0
[ ] ? handle_bug+0x6e/0xb0
[ ] ? exc_invalid_op+0x18/0x80
[ ] ? asm_exc_invalid_op+0x1b/0x20
[ ] ? xe_gt_sriov_vf_write32+0xc6/0x580 [xe]
[ ] ? xe_gt_sriov_vf_write32+0xc6/0x580 [xe]
[ ] ? xe_gt_tlb_invalidation_reset+0xef/0x110 [xe]
[ ] ? __mutex_unlock_slowpath+0x41/0x2e0
[ ] xe_mmio_write32+0x64/0x150 [xe]
[ ] do_gt_reset+0x2f/0xa0 [xe]
[ ] gt_reset_worker+0x14e/0x1e0 [xe]
[ ] process_one_work+0x21c/0x740
[ ] worker_thread+0x1db/0x3c0
```

Fix that by sending H2G VF_RESET(0x5507) action instead.

Closes: <https://gitlab.freedesktop.org/drm/xe/kernel/-/issues/4078>

Signed-off-by: Michal Wajdeczko <michal.wajdeczko@intel.com>

Reviewed-by: Lucas De Marchi <lucas.demarchi@intel.com>

Link: <https://patchwork.freedesktop.org/patch/msgid/20250131182502.852-1-michal.wajdeczko@intel.com>

Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

```
-rw-r--r-- drivers/gpu/drm/xe/xe_gt.c      4
-rw-r--r-- drivers/gpu/drm/xe/xe_gt_sriov_vf.c 16
-rw-r--r-- drivers/gpu/drm/xe/xe_gt_sriov_vf.h 1
```

3 files changed, 21 insertions, 0 deletions

```

diff --git a/drivers/gpu/drm/xe/xe_gt.c b/drivers/gpu/drm/xe/xe_gt.c
index 98fe8573e054e9..17ba15132a9840 100644
--- a/drivers/gpu/drm/xe/xe_gt.c
+++ b/drivers/gpu/drm/xe/xe_gt.c
@@ -32,6 +32,7 @@
#include "xe_gt_pagefault.h"
#include "xe_gt_printk.h"
#include "xe_gt_sriov_pf.h"
+include "xe_gt_sriov_vf.h"
#include "xe_gt_sysfs.h"
#include "xe_gt_tlb_invalidation.h"
#include "xe_gt_topology.h"
@@ -647,6 +648,9 @@ static int do_gt_reset(struct xe_gt *gt)
{
    int err;

+    if (IS_SRIOV_VF(gt_to_xe(gt)))
+        return xe_gt_sriov_vf_reset(gt);
+
    xe_gsc_wa_14015076503(gt, true);

    xe_mmio_write32(gt, GDRST, GRDOM_FULLL);

diff --git a/drivers/gpu/drm/xe/xe_gt_sriov_vf.c b/drivers/gpu/drm/xe/xe_gt_sriov_vf.c
index 4ebc82e607af65..f982d6f9f218d8 100644
--- a/drivers/gpu/drm/xe/xe_gt_sriov_vf.c
+++ b/drivers/gpu/drm/xe/xe_gt_sriov_vf.c
@@ -57,6 +57,22 @@ static int vf_reset_guc_state(struct xe_gt *gt)
    return err;
}

+/**
+ * xe_gt_sriov_vf_reset - Reset GuC VF internal state.
+ * @gt: the &xe_gt
+ *
+ * It requires functional `GuC MMIO based communication`.
+ *
+ * Return: 0 on success or a negative error code on failure.
+ */
+int xe_gt_sriov_vf_reset(struct xe_gt *gt)
+{
+    if (!xe_device_uc_enabled(gt_to_xe(gt)))
+        return -ENODEV;
+
+    return vf_reset_guc_state(gt);
+}
+
static int guc_action_match_version(struct xe_guc *guc,
                                   u32 wanted_branch, u32 wanted_major, u32 wanted_minor,
                                   u32 *branch, u32 *major, u32 *minor, u32 *patch)

diff --git a/drivers/gpu/drm/xe/xe_gt_sriov_vf.h b/drivers/gpu/drm/xe/xe_gt_sriov_vf.h
index e541ce57bec246..576ff5e795a8b0 100644
--- a/drivers/gpu/drm/xe/xe_gt_sriov_vf.h
+++ b/drivers/gpu/drm/xe/xe_gt_sriov_vf.h
@@ -12,6 +12,7 @@ struct drm_printer;
struct xe_gt;
struct xe_reg;

+int xe_gt_sriov_vf_reset(struct xe_gt *gt);
int xe_gt_sriov_vf_bootstrap(struct xe_gt *gt);
int xe_gt_sriov_vf_query_config(struct xe_gt *gt);
int xe_gt_sriov_vf_connect(struct xe_gt *gt);

```