author	Yunhui Cui <cuiyunhui@bytedance.com>	2025-02-20 20:17:16 +0800
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-04-20 10:22:24 +0200
commit	a71c6fc87b2b9905dc2e38887fe4122287216be9 (patch)	
tree	1c1949c16c0738339054d3ae356944a4105c793a	
parent	6ca553fe19053adb486c8d460604c4be3d6fb120 (diff)	
download	linux-a71c6fc87b2b9905dc2e38887fe4122287216be9.tar.gz	

perf/dwc_pcie: fix duplicate pci_dev devices

```
[ Upstream commit 7f35b429802a8065aa61e2a3f567089649f4d98e ]

During platform_device_register, wrongly using struct device
pci_dev as platform_data caused a kmemdup copy of pci_dev. Worse
still, accessing the duplicated device leads to list corruption as its
mutex content (e.g., list, magic) remains the same as the original.

Signed-off-by: Yunhui Cui <cuiyunhui@bytedance.com>
Reviewed-by: Shuai Xue <xueshuai@linux.alibaba.com>
Link: https://lore.kernel.org/r/20250220121716.50324-3-cuiyunhui@bytedance.com
Signed-off-by: Will Deacon <will@kernel.org>
Signed-off-by: Sasha Levin <sashal@kernel.org>
```

Diffstat

-rw-r--r-- drivers/perf/dwc_pcie_pmu.c 18

1 files changed, 12 insertions, 6 deletions

```
diff --git a/drivers/perf/dwc_pcie_pmu.c b/drivers/perf/dwc_pcie_pmu.c
index 19fa2ba8dd6704..f851e070760c56 100644
--- a/drivers/perf/dwc_pcie_pmu.c
+++ b/drivers/perf/dwc_pcie_pmu.c
@@ -565,9 +565,7 @@ static int dwc_pcie_register_dev(struct pci_dev *pdev)
        u32 sbdf;

        sbdf = (pci_domain_nr(pdev->bus) << 16) | PCI_DEVID(pdev->bus->number, pdev->devfn);
-       plat_dev = platform_device_register_data(NULL, "dwc_pcie_pmu", sbdf,
-                                                pdev, sizeof(*pdev));
-
+       plat_dev = platform_device_register_simple("dwc_pcie_pmu", sbdf, NULL, 0);
        if (IS_ERR(plat_dev))
                return PTR_ERR(plat_dev);

@@ -616,18 +614,26 @@ static struct notifier_block dwc_pcie_pmu_nb = {

 static int dwc_pcie_pmu_probe(struct platform_device *plat_dev)
 {
-       struct pci_dev *pdev = plat_dev->dev.platform_data;
+       struct pci_dev *pdev;
        struct dwc_pcie_pmu *pcie_pmu;
        char *name;
        u32 sbdf;
```

```
        u16 vsec;
        int ret;

+       sbdf = plat_dev->id;
+       pdev = pci_get_domain_bus_and_slot(sbdf >> 16, PCI_BUS_NUM(sbdf & 0xffff),
+                                          sbdf & 0xff);
+       if (!pdev) {
+               pr_err("No pdev found for the sbdf 0x%x\n", sbdf);
+               return -ENODEV;
+       }
+
        vsec = dwc_pcie_des_cap(pdev);
        if (!vsec)
                return -ENODEV;

-       sbdf = plat_dev->id;
+       pci_dev_put(pdev);
        name = devm_kasprintf(&plat_dev->dev, GFP_KERNEL, "dwc_rootport_%x", sbdf);
        if (!name)
                return -ENOMEM;
@@ -642,7 +648,7 @@ static int dwc_pcie_pmu_probe(struct platform_device *plat_dev)
        pcie_pmu->on_cpu = -1;
        pcie_pmu->pmu = (struct pmu){
                .name           = name,
-               .parent         = &pdev->dev,
+               .parent         = &plat_dev->dev,
                .module         = THIS_MODULE,
                .attr_groups    = dwc_pcie_attr_groups,
                .capabilities   = PERF_PMU_CAP_NO_EXCLUDE,
```