author	Suzuki K Poulose <suzuki.poulose@arm.com>	2025-04-22 17:16:16 +0100
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-05-02 07:50:46 +0200
commit	2f2803e4b5e4df2b08d378deaab78b1681ef9b30 (patch)	
tree	116586e4e39bc27fe2ab488acbd96709520e8b04	
parent	3d36fae38312d061279b42a3c2269ee372189bbc (diff)	
download	linux-2f2803e4b5e4df2b08d378deaab78b1681ef9b30.tar.gz	

diff options

context: 3

space: include

mode: unified

irqchip/gic-v2m: Prevent use after free of gicv2m_get_fwnode()

```
commit 3318dc299b072a0511d6dfd8367f3304fb6d9827 upstream.

With ACPI in place, gicv2m_get_fwnode() is registered with the pci
subsystem as pci_msi_get_fwnode_cb(), which may get invoked at runtime
during a PCI host bridge probe. But, the call back is wrongly marked as
__init, causing it to be freed, while being registered with the PCI
subsystem and could trigger:

 Unable to handle kernel paging request at virtual address ffff8000816c0400
  gicv2m_get_fwnode+0x0/0x58 (P)
  pci_set_bus_msi_domain+0x74/0x88
  pci_register_host_bridge+0x194/0x548

This is easily reproducible on a Juno board with ACPI boot.

Retain the function for later use.

Fixes: 0644b3daca28 ("irqchip/gic-v2m: acpi: Introducing GICv2m ACPI support")
Signed-off-by: Suzuki K Poulose <suzuki.poulose@arm.com>
Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
Signed-off-by: Ingo Molnar <mingo@kernel.org>
Reviewed-by: Marc Zyngier <maz@kernel.org>
Cc: stable@vger.kernel.org
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

Diffstat

-rw-r--r-- drivers/irqchip/irq-gic-v2m.c 2

1 files changed, 1 insertions, 1 deletions

```
diff --git a/drivers/irqchip/irq-gic-v2m.c b/drivers/irqchip/irq-gic-v2m.c
index d83c2c85962c37..683e8721e3b498 100644
--- a/drivers/irqchip/irq-gic-v2m.c
+++ b/drivers/irqchip/irq-gic-v2m.c
@@ -454,7 +454,7 @@ static int __init gicv2m_of_init(struct fwnode_handle *parent_handle,
 #ifdef CONFIG_ACPI
 static int acpi_num_msi;

-static __init struct fwnode_handle *gicv2m_get_fwnode(struct device *dev)
+static struct fwnode_handle *gicv2m_get_fwnode(struct device *dev)
 {
        struct v2m_data *data;
```