



author Lin Ma <linma@zju.edu.cn> 2023-03-07 23:29:17 +0800
committer Tony Nguyen <anthony.l.nguyen@intel.com> 2023-03-16 08:55:26 -0700
commit [65f69851e44d71248b952a687e44759a7abb5016](#) (patch)
tree [9e0b43261bb3ad2084f94e7a6fa665488c9b0172](#)
parent [cdd28833100c18a469c85a1cc3de9f6bbbe6caa0](#) (diff)
download [linux-65f69851e44d71248b952a687e44759a7abb5016.tar.gz](#)

diff options

context:
space:
mode:

igb: revert rtnl_lock() that causes deadlock

The commit 6faee3d4ee8b ("igb: Add lock to avoid data race") adds rtnl_lock to eliminate a false data race shown below

(FREE from device detaching)	(USE from netdev core)
igb_remove	igb_ndo_get_vf_config
igb_disable_sriov	vf >= adapter->vfs_allocated_count?
kfree(adapter->vf_data)	
adapter->vfs_allocated_count = 0	memcpy(... adapter->vf_data[vf]

The above race will never happen and the extra rtnl_lock causes deadlock below

```
[ 141.420169] <TASK>
[ 141.420672] __schedule+0x2dd/0x840
[ 141.421427] schedule+0x50/0xc0
[ 141.422041] schedule_preempt_disabled+0x11/0x20
[ 141.422678] __mutex_lock.isra.13+0x431/0x6b0
[ 141.423324] unregister_netdev+0xe/0x20
[ 141.423578] igbvf_remove+0x45/0xe0 [igbvf]
[ 141.423791] pci_device_remove+0x36/0xb0
[ 141.423990] device_release_driver_internal+0xc1/0x160
[ 141.424270] pci_stop_bus_device+0x6d/0x90
[ 141.424507] pci_stop_and_remove_bus_device+0xe/0x20
[ 141.424789] pci_iov_remove_virtfn+0xba/0x120
[ 141.425452] sriov_disable+0x2f/0xf0
[ 141.425679] igb_disable_sriov+0x4e/0x100 [igb]
[ 141.426353] igb_remove+0xa0/0x130 [igb]
[ 141.426599] pci_device_remove+0x36/0xb0
[ 141.426796] device_release_driver_internal+0xc1/0x160
[ 141.427060] driver_detach+0x44/0x90
[ 141.427253] bus_remove_driver+0x55/0xe0
[ 141.427477] pci_unregister_driver+0x2a/0xa0
[ 141.428296] __x64_sys_delete_module+0x141/0x2b0
[ 141.429126] ? mntput_no_expire+0x4a/0x240
[ 141.429363] ? syscall_trace_enter.isra.19+0x126/0x1a0
[ 141.429653] do_syscall_64+0x5b/0x80
[ 141.429847] ? exit_to_user_mode_prepare+0x14d/0x1c0
[ 141.430109] ? syscall_exit_to_user_mode+0x12/0x30
[ 141.430849] ? do_syscall_64+0x67/0x80
[ 141.431083] ? syscall_exit_to_user_mode_prepare+0x183/0x1b0
[ 141.431770] ? syscall_exit_to_user_mode+0x12/0x30
```

```
[ 141.432482] ? do_syscall_64+0x67/0x80
[ 141.432714] ? exc_page_fault+0x64/0x140
[ 141.432911] entry_SYSCALL_64_after_hwframe+0x72/0xdc
```

Since the `igb_disable_sriov()` will call `pci_disable_sriov()` before releasing any resources, the netdev core will synchronize the cleanup to avoid any races. This patch removes the useless `rtnl_(un)lock` to guarantee correctness.

CC: stable@vger.kernel.org

Fixes: 6faee3d4ee8b ("igb: Add lock to avoid data race")

Reported-by: Corinna Vinschen <vinschen@redhat.com>

Link: <https://lore.kernel.org/intel-wired-lan/ZAcJvkEPqWeJH02r@calimero.vinschen.de/>

Signed-off-by: Lin Ma <linma@zju.edu.cn>

Tested-by: Corinna Vinschen <vinschen@redhat.com>

Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>

Reviewed-by: Simon Horman <simon.horman@corigine.com>

Tested-by: Rafal Romanowski <rafal.romanowski@intel.com>

Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>

Diffstat

```
-rw-r--r-- drivers/net/ethernet/intel/igb/igb_main.c 2
```

1 files changed, 0 insertions, 2 deletions

```
diff --git a/drivers/net/ethernet/intel/igb/igb_main.c b/drivers/net/ethernet/intel/igb/igb_main.c
index 03bc1e8af575f7..5532361b0e9453 100644
```

```
--- a/drivers/net/ethernet/intel/igb/igb_main.c
```

```
+++ b/drivers/net/ethernet/intel/igb/igb_main.c
```

```
@@ -3863,9 +3863,7 @@ static void igb_remove(struct pci_dev *pdev)
     igb_release_hw_control(adapter);
```

```
 #ifdef CONFIG_PCI_IOV
```

```
-     rtnl_lock();
     igb_disable_sriov(pdev);
-     rtnl_unlock();
```

```
 #endif
```

```
     unregister_netdev(netdev);
```