Subject: [patch 0/1] fix veth netif_carrier_off
Posted by Daniel Lezcano on Tue, 18 Sep 2007 16:56:45 GMT
View Forum Message <> Reply to Message

When I tryed the veth driver, I fall into a kernel oops.

qemu login: Oops: 0000 [#1]
Modules linked in:
CPU: 0
EIP: 0060:[<c0265c9e>]   Not tainted VLI
EFLAGS: 00000202   (2.6.23-rc6-g754f885d-dirty #33)
EIP is at __linkwatch_run_queue+0x6a/0x175
eax: c7fc9550   ebx: 6b6b6b6b   ecx: c3360c80   edx: 00000246
esi: 00000001   edi: 6b6b6b6b   ebp: c7fd9f7c   esp: c7fd9f5c
ds: 007b   es: 007b   fs: 0000 gs: 0000 ss: 0068
Process events/0 (pid: 5, ti=c7fd8000 task=c7fc9550 task.ti=c7fd8000)
Stack: c7fee5a8 c0387680 c7fd9f74 c02e1aaa 4f732564 c0387684 c7fee5a8 c0387680
       c7fd9f84 c0265dc9 c7fd9fac c011fb3c c7fd9f94 c02e277e c7fd9fac c02e1166
       c0265da9 c7fee5a8 c0120203 c7fd9fc8 c7fd9fd0 c01202ba 00000000 c7fc9550
Call Trace:
 [<c0102c69>] show_trace_log_lvl+0x1a/0x2f
 [<c0102d1b>] show_stack_log_lvl+0x9d/0xa5
 [<c0102ee1>] show_registers+0x1be/0x28f
 [<c010309a>] die+0xe8/0x208
 [<c010d5a1>] do_page_fault+0x4ba/0x595
 [<c02e2842>] error_code+0x6a/0x70
 [<c0265dc9>] linkwatch_event+0x20/0x27
 [<c011fb3c>] run_workqueue+0x7c/0x102
 [<c01202ba>] worker_thread+0xb7/0xc5
 [<c012270c>] kthread+0x39/0x61
 [<c0102913>] kernel_thread_helper+0x7/0x10
 =======================
Code: b8 60 76 38 c0 e8 e3 ca 07 00 b8 60 76 38 c0 8b 1d 78 a7 3d c0 c7 05 78 a7 3d c0 00 00 00 00 e8 df ca 07 00 e9 ed 00 00 00 85 f6 <8b> bb f4 01 00 00 74 17 89 d8 e8 73 fe ff ff 85 c0 75 0c 89 d8
EIP: [<c0265c9e>] __linkwatch_run_queue+0x6a/0x175 SS:ESP 0068:c7fd9f5c
Slab corruption: size-2048 start=c473eac8, len=2048
Redzone: 0x9f911029d74e35b/0x9f911029d74e35b.
Last user: [<c025be72>](free_netdev+0x1f/0x41)
200: 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b c0 e2 73 c4
Prev obj: start=c473e2b0, len=2048
Redzone: 0xd84156c5635688c0/0xd84156c5635688c0.
Last user: [<c025bed0>](alloc_netdev_mq+0x3c/0xa1)
000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
010: 76 65 74 68 30 00 00 00 00 00 00 00 00 00 00 00
Next obj: start=c473f2e0, len=2048
Redzone: 0x9f911029d74e35b/0x9f911029d74e35b.
Last user: [<c0260e69>](neigh_sysctl_unregister+0x2b/0x2e)

000: 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b
010: 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b

That happens when trying to add the veth driver using the ip command:

ip link add veth0

which fail.

It appears that the netif_carrier_off is placed into the setup function
and this one is called before register_netdevice.

The register_netdevice function does a lot of initialization to the netdev
and if the netif_carrier_off is called before the register_netdev function,
it will use and trigger an event for an uninitialized netdev.

--

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

---

Subject: [patch 1/1] fix bad netif_carrier_off place
Posted by Daniel Lezcano on Tue, 18 Sep 2007 16:56:46 GMT
View Forum Message <> Reply to Message

From: Daniel Lezcano <dlezcano@fr.ibm.com>

If the netif_carrier_off is called before register_netdev
that will use and generate an event for a non initialized network
device and that leads to a Oops.

I moved the netif_carrier_off from the setup function after each
register_netdev call.

Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>
---
drivers/net/veth.c | 5 ++++-
1 file changed, 4 insertions(+), 1 deletion(-)

Index: net-2.6.24/drivers/net/veth.c
===
--- net-2.6.24.orig/drivers/net/veth.c
+++ net-2.6.24/drivers/net/veth.c
@@ -286,7 +286,6 @@ static void veth_setup(struct net_device
dev->features |= NETIF_F_LLTX;
dev->init = veth_dev_init;

```
  dev->destructor = veth_dev_free;
- netif_carrier_off(dev);
 }

 /*
@@ -357,6 +356,8 @@ static int veth_newlink(struct net_devic
  if (err < 0)
   goto err_register_peer;

+ netif_carrier_off(peer);
+
  /*
   * register dev last
   *
@@ -382,6 +383,8 @@ static int veth_newlink(struct net_devic
  if (err < 0)
   goto err_register_dev;

+ netif_carrier_off(dev);
+
  /*
   * tie the deviced together
   */

--
```

_____

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

---

Subject: Re: [patch 1/1] fix bad netif_carrier_off place
Posted by Pavel Emelianov on Wed, 19 Sep 2007 08:35:43 GMT
View Forum Message <> Reply to Message

> From: Daniel Lezcano <dlezcano@fr.ibm.com>
>
> If the netif_carrier_off is called before register_netdev
> that will use and generate an event for a non initialized network
> device and that leads to a Oops.
>
> I moved the netif_carrier_off from the setup function after each
> register_netdev call.
>
> Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>

Acked-by: Pavel Emelyanov <xemul@openvz.org>

Eric, if you managed to reproduce your oopses, does this patch help?

BTW, this problem exists for some other drivers, no? E.g.
xpnet_init() does the same thing and probably oopses as well.

> ---
> drivers/net/veth.c | 5 ++++-
> 1 file changed, 4 insertions(+), 1 deletion(-)
>
> Index: net-2.6.24/drivers/net/veth.c
> ===
> --- net-2.6.24.orig/drivers/net/veth.c
> +++ net-2.6.24/drivers/net/veth.c
> @@ -286,7 +286,6 @@ static void veth_setup(struct net_device
>   dev->features |= NETIF_F_LLTX;
>   dev->init = veth_dev_init;
>   dev->destructor = veth_dev_free;
> - netif_carrier_off(dev);
> }
>
> /*
> @@ -357,6 +356,8 @@ static int veth_newlink(struct net_devic
>   if (err < 0)
>   goto err_register_peer;
>
> + netif_carrier_off(peer);
> +
>   /*
>   * register dev last
>   *
> @@ -382,6 +383,8 @@ static int veth_newlink(struct net_devic
>   if (err < 0)
>   goto err_register_dev;
>
> + netif_carrier_off(dev);
> +
>   /*
>   * tie the deviced together
>   */
>
> --

_____

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers