Subject: [PATCH 2.6.18] e1000: fix initialization irqs
Posted by Mishin Dmitry on Wed, 22 Nov 2006 12:49:14 GMT
View Forum Message <> Reply to Message

In case of irqpoll boot option set, e1000 may oops due to:
1) e1000 register it's handler with e1000_request_irq
2) spurious interrupt happens
3) kernel tries to handle this interrupt with all available descs
4) e1000_intr is called and oops due to not initialized clean_rx handler,
because e1000_up is not called yet.

Solution is to initialize driver before handler registration.

Signed-off-by: Dmitry Mishin <dim@openvz.org>
Signed-off-by: Pavel Emelianov <xemul@openvz.org>

---
 e1000_main.c | 9 +++++----
 1 file changed, 5 insertions(+), 4 deletions(-)

--- ./drivers/net/e1000/e1000_main.c.e1000 2006-09-20 07:42:06.000000000 +0400
+++ ./drivers/net/e1000/e1000_main.c 2006-11-22 14:08:15.000000000 +0300
@@ -1206,14 +1206,15 @@ e1000_open(struct net_device *netdev)
  if ((err = e1000_setup_all_rx_resources(adapter)))
   goto err_setup_rx;

- err = e1000_request_irq(adapter);
- if (err)
- goto err_up;
-
  e1000_power_up_phy(adapter);

  if ((err = e1000_up(adapter)))
   goto err_up;
+
+ err = e1000_request_irq(adapter);
+ if (err)
+ goto err_up;
+
  adapter->mng_vlan_id = E1000_MNG_VLAN_NONE;
  if ((adapter->hw.mng_cookie.status &
      E1000_MNG_DHCP_COOKIE_STATUS_VLAN_SUPPORT)) {

Subject: Re: [PATCH 2.6.18] e1000: fix initialization irqs
Posted by Auke Kok on Wed, 22 Nov 2006 16:19:16 GMT

Dmitry Mishin wrote:
> In case of irqpoll boot option set, e1000 may oops due to:
> 1) e1000 register it's handler with e1000_request_irq
> 2) spurious interrupt happens
> 3) kernel tries to handle this interrupt with all available descs
> 4) e1000_intr is called and oops due to not initialized clean_rx handler,
> because e1000_up is not called yet.
>
> Solution is to initialize driver before handler registration.

I'm not so sure of that. even if we request our irq's to be routed it does not mean that we told the NIC to send them. It would suggest that you might be haunted by the MSI interrupts not working 100% correctly on some non-intel platform.

Now you're telling the NIC to send interrupts before we routed them I think.

If we _up() before we even have an interrupt, even more bad things could happen.

What is the reason you wrote this workaround?

Auke

>
> Signed-off-by: Dmitry Mishin <dim@openvz.org>
> Signed-off-by: Pavel Emelianov <xemul@openvz.org>
>
> ---
> e1000_main.c | 9 +++++----
> 1 file changed, 5 insertions(+), 4 deletions(-)
>
> ---
> --- ./drivers/net/e1000/e1000_main.c.e1000 2006-09-20 07:42:06.000000000 +0400
> +++ ./drivers/net/e1000/e1000_main.c 2006-11-22 14:08:15.000000000 +0300
> @@ -1206,14 +1206,15 @@ e1000_open(struct net_device *netdev)
>   if ((err = e1000_setup_all_rx_resources(adapter)))
>   goto err_setup_rx;
>
> - err = e1000_request_irq(adapter);
> - if (err)
> - goto err_up;
> -
>   e1000_power_up_phy(adapter);
>
>   if ((err = e1000_up(adapter)))
>   goto err_up;
> +
> + err = e1000_request_irq(adapter);

> + if (err)
> + goto err_up;
> +
>   adapter->mng_vlan_id = E1000_MNG_VLAN_NONE;
>   if ((adapter->hw.mng_cookie.status &
>   E1000_MNG_DHCP_COOKIE_STATUS_VLAN_SUPPORT)) {
> -
> To unsubscribe from this list: send the line "unsubscribe netdev" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html

---

Subject: Re: [PATCH 2.6.18] e1000: fix initialization irqs
Posted by Mishin Dmitry on Wed, 22 Nov 2006 16:43:52 GMT
View Forum Message <> Reply to Message

On Wednesday 22 November 2006 19:19, Auke Kok wrote:
> Dmitry Mishin wrote:
> > In case of irqpoll boot option set, e1000 may oops due to:
> > 1) e1000 register it's handler with e1000_request_irq
> > 2) spurious interrupt happens
> > 3) kernel tries to handle this interrupt with all available descs
> > 4) e1000_intr is called and oops due to not initialized clean_rx handler,
> > because e1000_up is not called yet.
> >
> > Solution is to initialize driver before handler registration.
>
> I'm not so sure of that. even if we request our irq's to be routed it does not mean that
> we told the NIC to send them. It would suggest that you might be haunted by the MSI
> interrupts not working 100% correctly on some non-intel platform.
>
> Now you're telling the NIC to send interrupts before we routed them I think.
>
> If we _up() before we even have an interrupt, even more bad things could happen.
>
> What is the reason you wrote this workaround?
The reason is simple, we've got this oops due to misrouted interrupts,
misrouted_irq() function. This workaround helps us, please, advice how to fix
it or fix it yourself.

>
> Auke
>
> >
> > Signed-off-by: Dmitry Mishin <dim@openvz.org>
> > Signed-off-by: Pavel Emelianov <xemul@openvz.org>
> >
> > ---

> > e1000_main.c | 9 +++++----
> > 1 file changed, 5 insertions(+), 4 deletions(-)
> >
> > ---
> > --- ./drivers/net/e1000/e1000_main.c.e1000 2006-09-20 07:42:06.000000000 +0400
> > +++ ./drivers/net/e1000/e1000_main.c 2006-11-22 14:08:15.000000000 +0300
> > @@ -1206,14 +1206,15 @@ e1000_open(struct net_device *netdev)
> >   if ((err = e1000_setup_all_rx_resources(adapter)))
> >   goto err_setup_rx;
> >
> > - err = e1000_request_irq(adapter);
> > - if (err)
> > - goto err_up;
> > -
> >   e1000_power_up_phy(adapter);
> >
> >   if ((err = e1000_up(adapter)))
> >   goto err_up;
> > +
> > + err = e1000_request_irq(adapter);
> > + if (err)
> > + goto err_up;
> > +
> >   adapter->mng_vlan_id = E1000_MNG_VLAN_NONE;
> >   if ((adapter->hw.mng_cookie.status &
> >   E1000_MNG_DHCP_COOKIE_STATUS_VLAN_SUPPORT)) {
> > -
> > To unsubscribe from this list: send the line "unsubscribe netdev" in
> > the body of a message to majordomo@vger.kernel.org
> > More majordomo info at http://vger.kernel.org/majordomo-info.html
> -
> To unsubscribe from this list: send the line "unsubscribe netdev" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html
>

--
Thanks,
Dmitry.

Subject: Re: [PATCH 2.6.18] e1000: fix initialization irqs
Posted by Auke Kok on Wed, 22 Nov 2006 20:51:19 GMT
View Forum Message <> Reply to Message

Dmitry Mishin wrote:
> On Wednesday 22 November 2006 19:19, Auke Kok wrote:
>> Now you're telling the NIC to send interrupts before we routed them I think.

>>
>> If we _up() before we even have an interrupt, even more bad things could happen.
>>
>> What is the reason you wrote this workaround?
> The reason is simple, we've got this oops due to misrouted interrupts,
> misrouted_irq() function. This workaround helps us, please, advice how to fix
> it or fix it yourself.

without knowing all the parameters that are involved, I cannot make an informed fix at all. I need to see the oops, a full dmesg dump, lspci -vv, and possibly more (preferably both before and after the OOPS).

If you can provide those then that would really help a lot, especially since the "fix" you provide introduces a great number of trap doors and other undetermined behaviour.

Cheers,

Auke
