

---

Subject: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [Pavel Emelianov](#) on Tue, 27 Nov 2007 16:21:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

This hook is protected with the RCU, so simple

```
if (br_should_route_hook)
    br_should_route_hook(...)
```

is not enough on some architectures.

Use the rcu\_dereference/rcu\_assign\_pointer in this case.

Fixed Stephen's comment concerning using the typeof().

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

---

```
diff --git a/net/bridge/br_input.c b/net/bridge/br_input.c
```

```
index 3cedd4e..0ee79a7 100644
```

```
--- a/net/bridge/br_input.c
```

```
+++ b/net/bridge/br_input.c
```

```
@@ -122,6 +122,7 @@ static inline int is_link_local(const unsigned char *dest)
    struct sk_buff *br_handle_frame(struct net_bridge_port *p, struct sk_buff *skb)
```

```
{
    const unsigned char *dest = eth_hdr(skb)->h_dest;
+ int (*rhook)(struct sk_buff *skb);
```

```
    if (!is_valid_ether_addr(eth_hdr(skb)->h_source))
        goto drop;
```

```
@@ -147,9 +148,9 @@ struct sk_buff *br_handle_frame(struct net_bridge_port *p, struct sk_buff
    *skb)
```

```
    switch (p->state) {
    case BR_STATE_FORWARDING:
```

```
-
```

```
- if (br_should_route_hook) {
```

```
- if (br_should_route_hook(skb))
```

```
+ rhook = rcu_dereference(br_should_route_hook);
```

```
+ if (rhook != NULL) {
```

```
+ if (rhook(skb))
```

```
    return skb;
```

```
    dest = eth_hdr(skb)->h_dest;
```

```
}
```

```
diff --git a/net/bridge/netfilter/ebtable_broute.c b/net/bridge/netfilter/ebtable_broute.c
```

```
index e44519e..be6f186 100644
```

```

--- a/net/bridge/netfilter/ebtable_broute.c
+++ b/net/bridge/netfilter/ebtable_broute.c
@@ -70,13 +70,13 @@ static int __init ebtable_broute_init(void)
    if (ret < 0)
        return ret;
    /* see br_input.c */
-   br_should_route_hook = ebt_broute;
+   rcu_assign_pointer(br_should_route_hook, ebt_broute);
    return ret;
}

static void __exit ebtable_broute_fini(void)
{
-   br_should_route_hook = NULL;
+   rcu_assign_pointer(br_should_route_hook, NULL);
    synchronize_net();
    ebt_unregister_table(&broute_table);
}

```

---

Subject: Re: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [Herbert Xu](#) on Thu, 29 Nov 2007 13:04:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Tue, Nov 27, 2007 at 07:21:08PM +0300, Pavel Emelyanov wrote:

```

> This hook is protected with the RCU, so simple
>
> if (br_should_route_hook)
>   br_should_route_hook(...)
>
> is not enough on some architectures.
>
> Use the rcu_dereference/rcu_assign_pointer in this case.
>
> Fixed Stephen's comment concerning using the typeof().
>
> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

```

Applied to net-2.6. Thanks Pavel!

```

> static void __exit ebtable_broute_fini(void)
> {
> - br_should_route_hook = NULL;
> + rcu_assign_pointer(br_should_route_hook, NULL);

```

Just for the record, rcu\_assign\_pointer is never necessary when  
you're assigning NULL. The reason is that rcu\_assign\_pointer serves

as a barrier between the initialisation of the content of what you're assigning and the actual assignment. Since NULL does not need to be initialised you don't need the barrier :)

Hmm, perhaps we could even build this logic into rcu\_assign\_pointer.

Then again, who still uses an Alpha? Mine died years ago :)

Cheers,

--

Visit Openswan at <http://www.openswan.org/>

Email: Herbert Xu ~{PmV>Hl~} <[herbert@gondor.apana.org.au](mailto:herbert@gondor.apana.org.au)>

Home Page: <http://gondor.apana.org.au/~herbert/>

PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

---

---

Subject: Re: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [paulmck](#) on Thu, 29 Nov 2007 14:36:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Fri, Nov 30, 2007 at 12:04:20AM +1100, Herbert Xu wrote:

> On Tue, Nov 27, 2007 at 07:21:08PM +0300, Pavel Emelyanov wrote:

> > This hook is protected with the RCU, so simple

> >

> > if (br\_should\_route\_hook)

> > br\_should\_route\_hook(...)

> >

> > is not enough on some architectures.

> >

> > Use the rcu\_dereference/rcu\_assign\_pointer in this case.

> >

> > Fixed Stephen's comment concerning using the typeof().

> >

> > Signed-off-by: Pavel Emelyanov <[xemul@openvz.org](mailto:xemul@openvz.org)>

>

> Applied to net-2.6. Thanks Pavel!

>

> > static void \_\_exit ebtable\_broute\_fini(void)

> > {

> > - br\_should\_route\_hook = NULL;

> > + rcu\_assign\_pointer(br\_should\_route\_hook, NULL);

>

> Just for the record, rcu\_assign\_pointer is never necessary when

> you're assigning NULL. The reason is that rcu\_assign\_pointer serves

> as a barrier between the initialisation of the content of what you're

> assigning and the actual assignment. Since NULL does not need to be

> initialised you don't need the barrier :)

Of course, if the `rcu_assign_pointer()` of `NULL` is not on a hot code path, the extra memory barrier might not be hurting enough to care.

> Hmm, perhaps we could even build this logic into `rcu_assign_pointer`.

That certainly is an interesting tradeoff... Save a memory barrier when assigning `NULL`, but pay an extra test and branch in all cases. Though it does make for a simpler rule -- just use `rcu_assign_pointer()` in all cases. Of course, if almost all `rcu_assign_pointer()` executions assign non-`NULL` pointers, the optimal strategy would be to leave the implementation of `rcu_assign_pointer()` alone, and simply enforce use of `rcu_assign_pointer()`, even if the pointer being assigned is `NULL`.

For a rough guess, if fewer than a few percent of `rcu_assign_pointer()` executions assign `NULL`, then it is best to simply change the rule. If more than about ten percent of `rcu_assign_pointer()` executions assign `NULL`, then it would make sense to put the check into the `rcu_assign_pointer()` primitive. The percentages would be of dynamic executions, rather than static counts of lines of code.

So, any intuitions on what fraction of the time `rcu_assign_pointer()` is assigning `NULL`? Failing that, what workload should be used to take the measurements? ;-)

> Then again, who still uses an Alpha? Mine died years ago :)

Although `rcu_dereference()` does a memory barrier only on Alpha, that of `rcu_assign_pointer()` is needed on any machine that does not preserve store order (Itanium, POWER, ARM, some MIPS boxes according to rumor, ...).

Thanx, Paul

> Cheers,

> --

> Visit Openswan at <http://www.openswan.org/>

> Email: Herbert Xu ~{PmV>Hl~} <[herbert@gondor.apana.org.au](mailto:herbert@gondor.apana.org.au)>

> Home Page: <http://gondor.apana.org.au/~herbert/>

> PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

> -

> To unsubscribe from this list: send the line "unsubscribe netdev" in

> the body of a message to [majordomo@vger.kernel.org](mailto:majordomo@vger.kernel.org)

> More majordomo info at <http://vger.kernel.org/majordomo-info.html>

---

Subject: Re: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [Herbert Xu](#) on Thu, 29 Nov 2007 23:49:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Thu, Nov 29, 2007 at 06:36:50AM -0800, Paul E. McKenney wrote:

>

> That certainly is an interesting tradeoff... Save a memory barrier  
> when assigning NULL, but pay an extra test and branch in all cases.  
> Though it does make for a simpler rule -- just use rcu\_assign\_pointer()  
> in all cases. Of course, if almost all rcu\_assign\_pointer() executions  
> assign non-NULL pointers, the optimal strategy would be to leave the  
> implementation of rcu\_assign\_pointer() alone, and simply enforce use  
> of rcu\_assign\_pointer(), even if the pointer being assigned is NULL.

I was thinking of something much simpler. If the second argument is constant and NULL, then skip the barrier. No run-time slow-down at all.

> Although rcu\_dereference() does a memory barrier only on Alpha, that of  
> rcu\_assign\_pointer() is needed on any machine that does not preserve store  
> order (Itanium, POWER, ARM, some MIPS boxes according to rumor, ...).

Good point!

Thanks,

--

Visit Openswan at <http://www.openswan.org/>

Email: Herbert Xu ~{PmV>Hl~} <[herbert@gondor.apana.org.au](mailto:herbert@gondor.apana.org.au)>

Home Page: <http://gondor.apana.org.au/~herbert/>

PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

---

---

Subject: Re: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [paulmck](#) on Fri, 30 Nov 2007 01:25:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Fri, Nov 30, 2007 at 10:49:00AM +1100, Herbert Xu wrote:

> On Thu, Nov 29, 2007 at 06:36:50AM -0800, Paul E. McKenney wrote:

> >

> > That certainly is an interesting tradeoff... Save a memory barrier  
> > when assigning NULL, but pay an extra test and branch in all cases.  
> > Though it does make for a simpler rule -- just use rcu\_assign\_pointer()  
> > in all cases. Of course, if almost all rcu\_assign\_pointer() executions  
> > assign non-NULL pointers, the optimal strategy would be to leave the  
> > implementation of rcu\_assign\_pointer() alone, and simply enforce use  
> > of rcu\_assign\_pointer(), even if the pointer being assigned is NULL.

>

> I was thinking of something much simpler. If the second argument is

> constant and NULL, then skip the barrier. No run-time slow-down at  
> all.

That certainly makes a lot of sense!!!

You have in mind something like the following?

```
#define rcu_assign_pointer(p, v) \
({ \
  if (!__builtin_constant_p(v) || \
      ((v) != NULL)) \
    smp_wmb(); \
  (p) = (v); \
})
```

If so, I will do some testing and submit a patch. Probably to Gautham's preemptible-RCU patchset to avoid gratuitously complicating his life, especially given that he very graciously agreed to take it over from me. We should be able to live with the overhead in the meantime. ;-)

Thanx, Paul

> > Although rcu\_dereference() does a memory barrier only on Alpha, that of  
> > rcu\_assign\_pointer() is needed on any machine that does not preserve store  
> > order (Itanium, POWER, ARM, some MIPS boxes according to rumor, ...).  
>  
> Good point!  
>  
> Thanks,  
> --  
> Visit Openswan at <http://www.openswan.org/>  
> Email: Herbert Xu ~{PmV>Hl~} <[herbert@gondor.apana.org.au](mailto:herbert@gondor.apana.org.au)>  
> Home Page: <http://gondor.apana.org.au/~herbert/>  
> PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

---

---

Subject: Re: [PATCH (resubmit)][BRIDGE] Properly dereference the  
br\_should\_route\_hook

Posted by [Herbert Xu](#) on Fri, 30 Nov 2007 02:31:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On Thu, Nov 29, 2007 at 05:25:01PM -0800, Paul E. McKenney wrote:

>  
> You have in mind something like the following?  
>  
> #define rcu\_assign\_pointer(p, v) \  
> ({ \  
> if (!\_\_builtin\_constant\_p(v) || \

```
> ((v) != NULL) \
> smp_wmb(); \
> (p) = (v); \
> })
```

Yes that's what I was thinking.

Thanks,

--

Visit Openswan at <http://www.openswan.org/>

Email: Herbert Xu ~{PmV>Hl~} <[herbert@gondor.apana.org.au](mailto:herbert@gondor.apana.org.au)>

Home Page: <http://gondor.apana.org.au/~herbert/>

PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

---