Subject: [PATCH] [IPV4]: Remove warning in node_set_parent.
Posted by den on Mon, 11 Feb 2008 08:47:17 GMT
View Forum Message <> Reply to Message

net/ipv4/fib_trie.c: In function 'node_set_parent':
net/ipv4/fib_trie.c:184: warning: assignment makes integer from pointer
without a cast

Signed-off-by: Denis V. Lunev <den@openvz.org>
---
 net/ipv4/fib_trie.c | 7 ++++---
 1 files changed, 4 insertions(+), 3 deletions(-)

diff --git a/net/ipv4/fib_trie.c b/net/ipv4/fib_trie.c
index f5fba3f..1753cd4 100644
--- a/net/ipv4/fib_trie.c
+++ b/net/ipv4/fib_trie.c
@@ -177,10 +177,11 @@ static inline struct tnode *node_parent_rcu(struct node *node)
  return rcu_dereference(ret);
 }

-static inline void node_set_parent(struct node *node, struct tnode *ptr)
+static inline void node_set_parent(struct node *node, struct tnode *__ptr)
 {
- rcu_assign_pointer(node->parent,
-      (unsigned long)ptr | NODE_TYPE(node));
+ struct node *ptr;
+ ptr = (struct node *)((unsigned long)__ptr | NODE_TYPE(node));
+ rcu_assign_pointer(node->parent, ptr);
 }

 static inline struct node *tnode_get_child(struct tnode *tn, unsigned int i)
--
1.5.3.rc5

Subject: Re: [PATCH] [IPV4]: Remove warning in node_set_parent.
Posted by Stephen Hemminger on Mon, 11 Feb 2008 19:48:49 GMT
View Forum Message <> Reply to Message

On Mon, 11 Feb 2008 11:47:17 +0300
"Denis V. Lunev" <den@openvz.org> wrote:

> net/ipv4/fib_trie.c: In function 'node_set_parent':
> net/ipv4/fib_trie.c:184: warning: assignment makes integer from pointer
> without a cast
>
> Signed-off-by: Denis V. Lunev <den@openvz.org>

> ---
> net/ipv4/fib_trie.c | 7 ++++---
> 1 files changed, 4 insertions(+), 3 deletions(-)
>
> diff --git a/net/ipv4/fib_trie.c b/net/ipv4/fib_trie.c
> index f5fba3f..1753cd4 100644
> --- a/net/ipv4/fib_trie.c
> +++ b/net/ipv4/fib_trie.c
> @@ -177,10 +177,11 @@ static inline struct tnode *node_parent_rcu(struct node *node)
>   return rcu_dereference(ret);
> }
>
> -static inline void node_set_parent(struct node *node, struct tnode *ptr)
> +static inline void node_set_parent(struct node *node, struct tnode *__ptr)
> {
> - rcu_assign_pointer(node->parent,
> -   (unsigned long)ptr | NODE_TYPE(node));
> + struct node *ptr;
> + ptr = (struct node *)((unsigned long)__ptr | NODE_TYPE(node));
> + rcu_assign_pointer(node->parent, ptr);
> }
>
> static inline struct node *tnode_get_child(struct tnode *tn, unsigned int i)

No, this causes new warning from assigning pointer (ptr) to integer node->parent.

Why not just change rcupdate.h to do the right thing.

>From a00f7cbf1c2f2282eced236e1e8b99b0fecd213a Mon Sep 17 00:00:00 2001
From: Stephen Hemminger <stephen.hemminger@vyatta.com>
Date: Mon, 11 Feb 2008 11:28:13 -0800
Subject: [PATCH] eliminate warnings when rcu_assign_pointer is used with unsigned long

It is reasonable to use RCU with non-pointer values, and describe
the optimization.

Signed-off-by: Stephen Hemminger <shemminger@vyatta.com>
---
 include/linux/rcupdate.h | 13 +++++++------
 1 files changed, 7 insertions(+), 6 deletions(-)

diff --git a/include/linux/rcupdate.h b/include/linux/rcupdate.h
index 37a642c..c44ac87 100644
--- a/include/linux/rcupdate.h
+++ b/include/linux/rcupdate.h
@@ -172,14 +172,15 @@ struct rcu_head {
  * structure after the pointer assignment. More importantly, this
  * call documents which pointers will be dereferenced by RCU read-side

```
  * code.
+ *
+ * If value is the NULL (constant 0), then no barrier is needed.
  */

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

 /**
--
1.5.3.8
```

--
Stephen Hemminger <stephen.hemminger@vyatta.com>

Subject: Re: [PATCH] [IPV4]: Remove warning in node_set_parent.
Posted by den on Tue, 12 Feb 2008 08:11:58 GMT
View Forum Message <> Reply to Message

ugly :), but

Acked-by: Denis V. Lunev <den@openvz.org>

On Mon, 2008-02-11 at 11:48 -0800, Stephen Hemminger wrote:
> On Mon, 11 Feb 2008 11:47:17 +0300
> "Denis V. Lunev" <den@openvz.org> wrote:
>
> > net/ipv4/fib_trie.c: In function 'node_set_parent':
> > net/ipv4/fib_trie.c:184: warning: assignment makes integer from pointer
> > without a cast
> >
> > Signed-off-by: Denis V. Lunev <den@openvz.org>
> > ---
> > net/ipv4/fib_trie.c | 7 ++++---

> > 1 files changed, 4 insertions(+), 3 deletions(-)
> >
> > diff --git a/net/ipv4/fib_trie.c b/net/ipv4/fib_trie.c
> > index f5fba3f..1753cd4 100644
> > --- a/net/ipv4/fib_trie.c
> > +++ b/net/ipv4/fib_trie.c
> > @@ -177,10 +177,11 @@ static inline struct tnode *node_parent_rcu(struct node *node)
> >   return rcu_dereference(ret);
> > }
> >
> > -static inline void node_set_parent(struct node *node, struct tnode *ptr)
> > +static inline void node_set_parent(struct node *node, struct tnode *__ptr)
> > {
> > - rcu_assign_pointer(node->parent,
> > -   (unsigned long)ptr | NODE_TYPE(node));
> > + struct node *ptr;
> > + ptr = (struct node *)((unsigned long)__ptr | NODE_TYPE(node));
> > + rcu_assign_pointer(node->parent, ptr);
> > }
> >
> > static inline struct node *tnode_get_child(struct tnode *tn, unsigned int i)
>
> No, this causes new warning from assigning pointer (ptr) to integer node->parent.
>
> Why not just change rcupdate.h to do the right thing.
>
> >From a00f7cbf1c2f2282eced236e1e8b99b0fecd213a Mon Sep 17 00:00:00 2001
> From: Stephen Hemminger <stephen.hemminger@vyatta.com>
> Date: Mon, 11 Feb 2008 11:28:13 -0800
> Subject: [PATCH] eliminate warnings when rcu_assign_pointer is used with unsigned long
>
> It is reasonable to use RCU with non-pointer values, and describe
> the optimization.
>
> Signed-off-by: Stephen Hemminger <shemminger@vyatta.com>
> ---
> include/linux/rcupdate.h | 13 +++++++------
> 1 files changed, 7 insertions(+), 6 deletions(-)
>
> diff --git a/include/linux/rcupdate.h b/include/linux/rcupdate.h
> index 37a642c..c44ac87 100644
> --- a/include/linux/rcupdate.h
> +++ b/include/linux/rcupdate.h
> @@ -172,14 +172,15 @@ struct rcu_head {
>   * structure after the pointer assignment. More importantly, this
>   * call documents which pointers will be dereferenced by RCU read-side
>   * code.
> + *

```
> + * If value is the NULL (constant 0), then no barrier is needed.
>   */
>
> -#define rcu_assign_pointer(p, v) \
> - ({ \
> -  if (!__builtin_constant_p(v) || \
> -      ((v) != NULL)) \
> -   smp_wmb(); \
> -  (p) = (v); \
> +#define rcu_assign_pointer(p, v)   \
> + ({      \
> +  if (!(__builtin_constant_p(v) && v)) \
> +   smp_wmb();   \
> +  (p) = (v);    \
>   })
>
>  /**
> --
> 1.5.3.8
>
>
>
>
```