Subject: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by Pavel Emelianov on Thu, 18 Oct 2007 11:53:52 GMT
View Forum Message <> Reply to Message

This routine scans the ipv6_fl_list whose update is
protected with the socket lock and the ip6_sk_fl_lock.

Since the socket lock is not taken in the lookup, use
the other one.

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

```
diff --git a/net/ipv6/ip6_flowlabel.c b/net/ipv6/ip6_flowlabel.c
index 8550df2..f40a086 100644
--- a/net/ipv6/ip6_flowlabel.c
+++ b/net/ipv6/ip6_flowlabel.c
@@ -190,14 +190,17 @@ struct ip6_flowlabel * fl6_sock_lookup(struct sock *sk, __be32 label)

  label &= IPV6_FLOWLABEL_MASK;

+ read_lock_bh(&ip6_sk_fl_lock);
  for (sfl=np->ipv6_fl_list; sfl; sfl = sfl->next) {
   struct ip6_flowlabel *fl = sfl->fl;
   if (fl->label == label) {
+   read_unlock_bh(&ip6_sk_fl_lock);
    fl->lastuse = jiffies;
    atomic_inc(&fl->users);
    return fl;
   }
  }
+ read_unlock_bh(&ip6_sk_fl_lock);
  return NULL;
 }
```

Subject: Re: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by yoshfuji on Thu, 18 Oct 2007 12:00:43 GMT
View Forum Message <> Reply to Message

In article <47174950.6060409@openvz.org> (at Thu, 18 Oct 2007 15:53:52 +0400), Pavel Emelyanov <xemul@openvz.org> says:

> This routine scans the ipv6_fl_list whose update is
> protected with the socket lock and the ip6_sk_fl_lock.

>   struct ip6_flowlabel *fl = sfl->fl;

```
>    if (fl->label == label) {
> +   read_unlock_bh(&ip6_sk_fl_lock);
>     fl->lastuse = jiffies;
>     atomic_inc(&fl->users);
>     return fl;
```

We should increment fl->users within the critical section, shouldn't we?

--yoshfuji

Subject: Re: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by Pavel Emelianov on Thu, 18 Oct 2007 12:11:58 GMT

View Forum Message <> Reply to Message

YOSHIFUJI Hideaki wrote:
> In article <47174950.6060409@openvz.org> (at Thu, 18 Oct 2007 15:53:52 +0400), Pavel Emelyanov <xemul@openvz.org> says:
>
>> This routine scans the ipv6_fl_list whose update is
>> protected with the socket lock and the ip6_sk_fl_lock.
>

```
>>    struct ip6_flowlabel *fl = sfl->fl;
>>    if (fl->label == label) {
>> +   read_unlock_bh(&ip6_sk_fl_lock);
>>     fl->lastuse = jiffies;
>>     atomic_inc(&fl->users);
>>     return fl;
>
```

> We should increment fl->users within the critical section, shouldn't we?

Not necessary. The users is more than zero (because it is
linked in the sock's list) so garbage collector won't catch
it in any way.

Thanks,
Pavel

> --yoshfuji
> -
> 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 2/3] Lost locking in fl6_sock_lookup

Posted by davem on Thu, 18 Oct 2007 12:14:26 GMT
View Forum Message <> Reply to Message

From: Pavel Emelyanov <xemul@openvz.org>
Date: Thu, 18 Oct 2007 16:11:58 +0400

> YOSHIFUJI Hideaki wrote:
> > In article <47174950.6060409@openvz.org> (at Thu, 18 Oct 2007 15:53:52 +0400), Pavel Emelyanov <xemul@openvz.org> says:
> >
> >> This routine scans the ipv6_fl_list whose update is
> >> protected with the socket lock and the ip6_sk_fl_lock.
> >
> >>   struct ip6_flowlabel *fl = sfl->fl;
> >>   if (fl->label == label) {
> >> +   read_unlock_bh(&ip6_sk_fl_lock);
> >>   fl->lastuse = jiffies;
> >>   atomic_inc(&fl->users);
> >>   return fl;
> >
> > We should increment fl->users within the critical section, shouldn't we?
>
> Not necessary. The users is more than zero (because it is
> linked in the sock's list) so garbage collector won't catch
> it in any way.

Right, we're grabbing an "extra" reference here and only
someone who gets the socket lock (which we have) can unlink
it and thus potentially drop the count to zero.

Subject: Re: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by davem on Thu, 18 Oct 2007 12:16:47 GMT
View Forum Message <> Reply to Message

From: Pavel Emelyanov <xemul@openvz.org>
Date: Thu, 18 Oct 2007 15:53:52 +0400

> This routine scans the ipv6_fl_list whose update is
> protected with the socket lock and the ip6_sk_fl_lock.
>
> Since the socket lock is not taken in the lookup, use
> the other one.
>
> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

Applied.

But I notice that I was wrong in my email, we don't
hold the socket lock here.

What prevents an unlink from the socket's list
and thus a reference count of zero occurring for
a brief moment?

Subject: Re: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by Pavel Emelianov on Thu, 18 Oct 2007 12:22:49 GMT
View Forum Message <> Reply to Message

David Miller wrote:
> From: Pavel Emelyanov <xemul@openvz.org>
> Date: Thu, 18 Oct 2007 15:53:52 +0400
>
>> This routine scans the ipv6_fl_list whose update is
>> protected with the socket lock and the ip6_sk_fl_lock.
>>
>> Since the socket lock is not taken in the lookup, use
>> the other one.
>>
>> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>
>
> Applied.
>
> But I notice that I was wrong in my email, we don't
> hold the socket lock here.
>
> What prevents an unlink from the socket's list
> and thus a reference count of zero occurring for
> a brief moment?

Oops. You're right here :(I looked at the ip6_fl_lock
and messed it with the ip6_sk_fl_lock.

Should I resend the whole patch, or just make an
incremental one?

Thanks,
Pavel

Subject: Re: [PATCH 2/3] Lost locking in fl6_sock_lookup
Posted by davem on Thu, 18 Oct 2007 12:33:46 GMT
View Forum Message <> Reply to Message

From: Pavel Emelyanov <xemul@openvz.org>
Date: Thu, 18 Oct 2007 16:22:49 +0400

> Oops. You're right here :(I looked at the ip6_fl_lock
> and messed it with the ip6_sk_fl_lock.
>
> Should I resend the whole patch, or just make an
> incremental one?

Please make an incremental one.

And hurry, I'm trying to go to bed :-)))