

---

Subject: Re: [patch 2/3][NETNS45][V2] make timewait unhash lock free  
Posted by [Daniel Lezcano](#) on Thu, 27 Sep 2007 13:09:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Denis V. Lunev wrote:

> Sorry for a delay in answer. A was ill last three days.  
> Some stylistic comments inside

>

> Daniel Lezcano wrote:

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

>>

>> The network namespace cleanup will remove all timewait sockets  
>> related to it because there are pointless.

>>

>> The problem is we need to browse the established hash table and  
>> for that we need to take the lock. For each timesocket we call  
>> inet\_deschedule and this one take the established hash table lock  
>> too.

>>

>> The following patchset split the removing of the established hash  
>> into two parts, one removing the node from the hash and another  
>> taking the lock and calling the first one.

>>

>> The network namespace cleanup can be done calling the lock free  
>> function.

>>

>> Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>

>> ---

>> include/net/inet\_timewait\_sock.h | 13 ++++++++  
>> net/ipv4/inet\_timewait\_sock.c | 40 ++++++++-----  
>> 2 files changed, 41 insertions(+), 12 deletions(-)

>>

>> Index: linux-2.6-netns/net/ipv4/inet\_timewait\_sock.c

>> =====

>> --- linux-2.6-netns.orig/net/ipv4/inet\_timewait\_sock.c

>> +++ linux-2.6-netns/net/ipv4/inet\_timewait\_sock.c

>> @@ -13,25 +13,28 @@

>> #include <net/inet\_timewait\_sock.h>

>> #include <net/ip.h>

>>

>> /\* Must be called with locally disabled BHs. \*/

>> -static void \_\_inet\_twsk\_kill(struct inet\_timewait\_sock \*tw,

>> - struct inet\_hashinfo \*hashinfo)

>> +static inline int inet\_twsk\_unhash(struct inet\_timewait\_sock \*tw,

>> + struct inet\_hashinfo \*hashinfo)

>> {

>> - struct inet\_bind\_hashbucket \*bhead;

>> - struct inet\_bind\_bucket \*tb;

```

>> - /* Unlink from established hashes. */
>> - struct inet_eshash_bucket *ehead = inet_eshash_bucket(hashinfo, tw->tw_hash);
>> + struct inet_eshash_bucket *ehead =
>> + inet_eshash_bucket(hashinfo, tw->tw_hash);
>>
>> write_lock(&ehead->lock);
>> - if (hlist_unhashed(&tw->tw_node)) {
>> + if (__inet_twsd_unhash(tw)) {
>> write_unlock(&ehead->lock);
>> - return;
>> + return 1;
>> }
>> - __hlist_del(&tw->tw_node);
>> - sk_node_init(&tw->tw_node);
>> write_unlock(&ehead->lock);
>>
>> - /* Disassociate with bind bucket. */
>> + return 0;
>> +}
> as far as I can understand the code, it will look better as below
>
> struct inet_eshash_bucket *ehead =
> inet_eshash_bucket(hashinfo, tw->tw_hash);
> int ret;
>
> write_lock(&ehead->lock);
> ret = __inet_twsd_unhash(tw);
> write_unlock(&ehead->lock);
> return ret;

```

Right. Will fix that. Thanks.

[ cut ]

```

>> +
>>> see above about inet_twsd_unhash. We should insert
> /* Disassociate with bind bucket. */
>>> here
>> + return 0;
>> +}

```

Is this comment for me ?

---

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

---