Subject: Re: [PATCH] Fix inet_diag.ko register vs rcv race
Posted by Herbert Xu on Thu, 29 Nov 2007 12:37:34 GMT
View Forum Message <> Reply to Message

On Tue, Nov 27, 2007 at 04:09:43PM +0300, Pavel Emelyanov wrote:

> The following race is possible when one cpu unregisters the handler
> while other one is trying to receive a message and call this one:

Good catch! But I think we need a bit more to close this fully.

Dumps can resume asynchronously which means that they won't be holding inet_diag_mutex. We can fix that pretty easily by giving that as our cb_mutex.

So could you add that to your patch and resubmit?

Arnaldo, synchronize_rcu() doesn't work on its own. Whoever accesses the object that it's supposed to protect has to use the correct RCU primitives for this to work.

Synchronisation is like tango, it always takes two to make it work :)

Thanks,
--
Visit Openswan at http://www.openswan.org/
Email: Herbert Xu ~{PmV>HI~} <herbert@gondor.apana.org.au>
Home Page: http://gondor.apana.org.au/~herbert/
PGP Key: http://gondor.apana.org.au/~herbert/pubkey.txt