

---

Subject: Re: [PATCH 5/5] make netlink user -> kernel interface synchronous  
Posted by [davem](#) on Thu, 11 Oct 2007 04:15:38 GMT

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

---

From: "Denis V. Lunev" <den@openvz.org>

Date: Fri, 5 Oct 2007 18:48:44 +0400

> This patch make processing netlink user -> kernel messages synchronous.  
> This change was inspired by the talk with Alexey Kuznetsov about current  
> netlink messages processing. He says that he was badly wrong when introduced  
> asynchronous user -> kernel communication.  
>  
> The call netlink\_unicast is the only path to send message to the kernel  
> netlink socket. But, unfortunately, it is also used to send data to the  
> user.  
>  
> Before this change the user message has been attached to the socket queue  
> and sk->sk\_data\_ready was called. The process has been blocked until all  
> pending messages were processed. The bad thing is that this processing  
> may occur in the arbitrary process context.  
>  
> This patch changes nlk->data\_ready callback to get 1 skb and force packet  
> processing right in the netlink\_unicast.  
>  
> Kernel -> user path in netlink\_unicast remains untouched.  
>  
> EINTR processing for in netlink\_run\_queue was changed. It forces rtnl\_lock  
> drop, but the process remains in the cycle until the message will be fully  
> processed. So, there is no need to use this kludges now.  
>  
> Signed-off-by: Denis V. Lunev <den@openvz.org>  
> Aacked-by: Alexey Kuznetsov <kuznet@ms2.inr.ac.ru>

Applied.

---