

---

Subject: Re: cleanup in workq and dst\_destroy  
Posted by [den](#) on Fri, 16 Nov 2007 17:06:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Daniel Lezcano wrote:

> Hi all,  
>  
> while doing ipv6 namespace, we were faced to a problem with the loopback  
> and the dst\_destroy function.  
>  
> When the network namespace exits, the cleanup function is called by  
> schedule\_work and this function will browse the net ops list to call the  
> different exit methods for the registered subsystems.  
>  
> The different subsystems will shutdown their resources and in particular  
> addrconf subsystem will ifdown the loopback. This function will call  
> rt6\_ifdown  
> -> fib6\_clean\_all  
> -> fib6\_clean\_node  
> -> fib6\_clean\_tree  
> -> fib6\_clean\_node  
> -> fib6\_del  
> -> fib6\_del\_route  
> -> rt6\_release  
> -> dst\_free  
> -> \_\_dst\_free  
>  
> The \_\_dst\_free function will schedule\_delayed\_work the dst\_gc\_work  
> function.  
>  
> The dst\_gc\_work will call dst\_destroy and finally this one will call  
> dst->ops->destroy ops function which is ip6\_dst\_destroy.  
>  
> The problem here is we have the workq blocked because we are running  
> inside the netns cleanup function. So the delayed work will not run  
> until we exits the cleanup function. But the loopback is still  
> referenced by the ip6 routes, the netdev\_unregister will loop  
> indefinitely => dead lock.  
>  
> By the way, this bug appears with ipv6 but it is perhaps pending with ipv4.  
>  
> Benjamin as proposed to create a separate workq for the network  
> namespace, so in the worst case we have the unregister looping until the  
> ip6 route are shut downed. Is it an acceptable solution ?  
>

we are doing this stuff in the special thread. There are a lot of  
difficult things to perform like synchronize\_net & netdev\_run\_todo inside

Regards,  
Den

---

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

---