

---

Subject: Re: [PATCH] usbatm: Update to use the kthread api.  
Posted by [ebiederm](#) on Fri, 15 Dec 2006 10:54:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Duncan Sands <[baldrick@free.fr](mailto:baldrick@free.fr)> writes:

> Hi Eric,  
>  
> presumably the problem is that if the thread has spontaneously exited, and  
> afterwards disconnect calls kthread\_stop, then things go boom. The same  
> problem exists (though with lesser consequences) when sending a signal.  
> There is already code in usbatm to avoid this problem with signals. Why  
> not just recycle it in the kthread\_stop case? I guess there is no  
> problem if you can guarantee that the following occurs:  
> if kthread\_stop is ever called for the kthread, then the kthread only  
> exits after seeing kthread\_should\_stop return true.

I suspect we can recycle the locking on the signal sending code. At least as a first pass. I have almost digested the problem sufficiently to write some code. Maybe this weekend.

>> To be clear I have a problem with using numeric pids of kernel threads,  
>  
> Yes, this is a problem with usbatm at the moment.  
>  
>> and with spawning threads from a possibly user space environment.  
>  
> Not the case with usbatm. It is always spawned from khubd.

That is where I thought we were at, doing the conversion so it is obvious and we can remove the use of kernel\_thread and daemonize would certainly be good. The more shared infrastructure we can reasonably have the more likely the code will function correctly.

Eric

---

Containers mailing list  
[Containers@lists.osdl.org](mailto:Containers@lists.osdl.org)  
<https://lists.osdl.org/mailman/listinfo/containers>

---