

---

Subject: Re: [PATCH] usbarm: Update to use the kthread api.  
Posted by [Duncan Sands](#) on Wed, 03 Jan 2007 19:12:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Friday 15 December 2006 11:45, Eric W. Biederman wrote:  
> Christoph Hellwig <hch@infradead.org> writes:  
>  
> > On Fri, Dec 15, 2006 at 03:17:57AM -0700, Eric W. Biederman wrote:  
> >  
> > I think the right fix is to enhance the kthread infrastructure to  
> > gracefully handle the case where the thread has stopped by itself  
> > and doesn't exist anymore at the time where we call kthread\_stop.  
>  
> Yep that is about where I thought we were at. Now we need to figure out  
> how to do that cleanly, and sanely.

There's a completely different solution, which is to use a workqueue instead of a kthread, with users providing a cancellation method. Recall that the functionality is provided by usbarm to drivers: they use it to perform slow initialization that is too slow to be done in the probe method. They register with usbarm, providing a "heavy\_init" method. They could also provide a "heavy\_cancel" method. (Any special data that heavy\_cancel needs can be stored in the existing driver private data structure; this structure is already passed to heavy\_init). In the case of the speedtch driver, it could place a completion in its private data structure; heavy\_cancel would just complete the completion. Rather than doing interruptible sleeps, it can use wait\_for\_completion\_timeout.

The only thing that worries me about this solution is... that you can't shoot down firmware loading from userspace anymore. For example, if heavy\_init is blocked loading firmware when the system is halted, it presumably won't react to the kill signal. Perhaps it is unimportant; and if not, I guess I can just re-enable signals in heavy\_init.

Ciao,

Duncan.

---

Containers mailing list  
Containers@lists.osdl.org  
<https://lists.osdl.org/mailman/listinfo/containers>

---