Subject: Re: [PATCH] kthread: saa7134-tvaudio.c
Posted by Cedric Le Goater on Wed, 30 Aug 2006 16:18:55 GMT
View Forum Message <> Reply to Message

Eric W. Biederman wrote:

[...]

>>> That plus the obvious bit. For the pid namespace we have to declare
>>> war on people storing a pid_t values. Either converting them to
>>> struct pid * or removing them entirely. Doing the kernel_thread to
>>> kthread conversion removes them entirely.
>> we've started that war, won a few battles but some drivers need more work
>> that a simple replace. If we could give some priorities, it would help to
>> focus on the most important ones. check out the list bellow.
>
> Sure, I think I can help.
>
> There are a couple of test I can think of that should help.
> 1) Is the pid value stored. If not a pid namespace won't affect
>   it's normal operation.

I've extracted this list from a table which includes a pid cache column.
this pid cache column is not complete yet. I'd be nice if we could use a
wiki to maintain this table, the existing openvz or vserver wiki ?

> 2) Is this thread started during kernel boot before this thread
>   could have a user space parent. If it can't have a user space
>   parent then it can't take a reference to user space resources.

ok we need to add this one.

> 3) Can the code be compiled modular and will it break when we stop
>   exporting kernel_thread.

got that also.

> 4) How frequently is this thing used. The more common code is probably
>   in better shape and more likely to get a good maintainer response, and
>   we care more :)

sure :) some drivers are for some exotic piece of hardware that are not
currently found on a standard server.

> irqbalanced from arch/i386/kernel/io_apic.c should be safe to leave alone
> because it doesn't store a pid_t, it is started during boot, and it can't
> be compiled modular.
>

>>From what I have seen you can shorten the list by several entries by removing
> code like irqbalanced that can't possibly cause us any problems.
> kvoyagerd from arch/i386/mach-voyager/voyager_thread.c is another one.

ok thanks, will update.

> The first on my personal hit list is nfs.
>> fs/lockd/clntlock.c
>> fs/nfs/delegation.c
>> net/sunrpc/svc.c
>
> Because it does store pid_t values, it isn't started during kernel boot,
> it can be compiled modular, and people use it all of the time.

yes yes. hard stuff though which requires time.

> I do agree from what I have seen, that changing idioms to the kthread way of
> doing things isn't simply a matter of substitute and replace which is
> unfortunate. Although the biggest hurdle seems to be to teach kernel threads
> to communicate with something besides signals. Which is a general help anyway.
>
> Unfortunately I'm distracted at the moment so I haven't gone through the entire
> list but I hope this helps.

we would need a wiki to maintain the work in progress on that topic while
we work on the pidspace.

another list to maintain would be the pid_t to struct pid replacement.

C.

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