
Subject: Re: [PATCH] task containersv11 add tasks file interface fix for cpusets
Posted by [Paul Jackson](#) on Thu, 11 Oct 2007 22:03:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

Several days ago, Paul M replied to Paul J:

> > Hmmm ... guess I'd have to loop over the cgroup twice, once to count
> > them (the 'count' field is not claimed to be accurate) and then again,
> > after I've kmalloc'd the tasks[] array, filling in the tasks[] array.
> >
> > On a big cgroup on a big system, this could easily be thousands of
> > iteration loops.
>
> But if userspace has to do it, the effect will be far more expensive.

Ah - but I'm not trying to optimize this particular operation (changing a cpusets 'cpus'). It's not at all performance critical.

I'm trying to minimize the amount of special purpose code in the kernel.

The maintenance costs of a line of kernel code are quite a bit higher than for a line of user code. I work hard to have most of my lines of kernel code be on well traveled code paths, of general usefulness, even if this means that some infrequent operations require yet more user source code lines and user CPU cycles, in order to be refactored as the combination of multiple system call primitives.

... all within reasonable limits, of course.

Corner case, special situation, non-trivial chunks of kernel code are very expensive. They don't get very good testing coverage in the real world, and end up harboring latent bugs for months or years, by which time it can be expensive to deal with them.

Be that as it may, I've just started digesting the actual code suggestions posted by yourself and David (thanks!) this last week. I just couldn't resist a bit of philosophizing ... sorry.

--

I won't rest till it's the best ...
Programmer, Linux Scalability
Paul Jackson <pj@sgi.com> 1.925.600.0401

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