
Subject: Re: [ckrm-tech] [PATCH 1/7] containers (V7): Generic container system abstracted from cpusets code

Posted by [Paul Jackson](#) on Sat, 24 Mar 2007 19:25:59 GMT

[View Forum Message](#) <> [Reply to Message](#)

> IMO, we need to use task_lock() in container_exit() to avoid this race.
>
> (I think this race already exists in mainline cpuset.c?)
>
> P.S : cpuset.c checks for PF_EXITING twice in attach_task(), while this
> patch seems to be checking only once. Is that fine?

I think the cpuset code is ok, because, as you note, it locks the task, picks off the cpuset pointer, and then checks a second time that the task still does not have PF_EXITING set:

In the kernel/cpuset.c code for attach_task():

```
task_lock(tsk);
oldcs = tsk->cpuset;
/*
 * After getting 'oldcs' cpuset ptr, be sure still not exiting.
 * If 'oldcs' might be the top_cpuset due to the_top_cpuset_hack
 * then fail this attach_task(), to avoid breaking top_cpuset.count.
 */
if (tsk->flags & PF_EXITING) {
    task_unlock(tsk);
    mutex_unlock(&callback_mutex);
    put_task_struct(tsk);
    return -ESRCH;
}
```

--

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