Subject: Re: [PATCH] task containersv11 add tasks file interface fix for cpusets Posted by David Rientjes on Sun, 07 Oct 2007 06:15:14 GMT

View Forum Message <> Reply to Message

On Sat, 6 Oct 2007, Paul Menage wrote:

- > > The getting and putting of the tasks will prevent them from exiting or
- > > being deallocated prematurely. But this is also a critical section that
- > > will need to be protected by some mutex so it doesn't race with other
- >> set cpus allowed().

>

- > Is that necessary? If some other process calls set cpus allowed()
- > concurrently with a cpuset cpus update, it's not clear that there's
- > any defined serialization semantics that have to be achieved, as long
- > as the end result is that the task's cpus_allowed are within the
- > cpuset's cpus_allowed.

>

It can race with sched_setaffinity(). It has to give up tasklist_lock as well to call set_cpus_allowed() and can race

```
cpus_allowed = cpuset_cpus_allowed(p);
cpus_and(new_mask, new_mask, cpus_allowed);
retval = set_cpus_allowed(p, new_mask);
```

and allow a task to have a cpu outside of the cpuset's new cpus_allowed if you've taken it away between cpuset_cpus_allowed() and set_cpus_allowed().

David

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