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

Posted by [Srivatsa Vaddagiri](#) on Wed, 07 Mar 2007 12:21:17 GMT

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

On Mon, Feb 12, 2007 at 12:15:22AM -0800, menage@google.com wrote:

> +/**

> + * container_lock - lock out any changes to container structures

> + *

> + * The out of memory (oom) code needs to mutex_lock containers

> + * from being changed while it scans the tasklist looking for a

> + * task in an overlapping container.

Which specific portion of oom code cares abt container structure being intact?

If I understand correctly, only cpuset requires this double locking. More specifically, cpusets cares about walking cpuset->parent list safely with callback_mutex held correct?

If that is the case, I think we can push container_lock entirely inside cpuset.c and not have others exposed to this double-lock complexity. This is possible because cpuset.c (build on top of containers) still has cpuset->parent and walking cpuset->parent list safely can be made possible with a second lock which is local to only cpuset.c.

--

Regards,
vatsa
