
Subject: Re: [PATCH 0/9] Containers (V9): Generic Process Containers

Posted by [Christoph Hellwig](#) on Mon, 30 Apr 2007 17:23:52 GMT

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

On Mon, Apr 30, 2007 at 10:42:25PM +0530, Srivatsa Vaddagiri wrote:

> On Sun, Apr 29, 2007 at 02:37:21AM -0700, Paul Jackson wrote:

> > It builds and boots and mounts the cpuset file system ok.

> > But trying to write the 'mems' file hangs the system hard.

>

> Basically we are attempting a read_lock(&tasklist_lock) in

> container_task_count() after taking write_lock_irq(&tasklist_lock) in

> update_nodemask()!

>

> This patch seems to fix the prb for me:

>

>

> Fix write_lock() followed by read_lock() bug by introducing a 2nd

> argument to be passed into container_task_count. Other choice is to

> introduce a lock and unlocked versions of container_task_count() ..

>

> Signed-off-by : Srivatsa Vaddagiri <vatsa@in.ibm.com>

> -int container_task_count(const struct container *cont) {

> +int container_task_count(const struct container *cont, int take_lock) {

> int count = 0;

> struct task_struct *g, *p;

> struct container_subsys_state *css;

> int subsys_id;

> get_first_subsys(cont, &css, &subsys_id);

>

> - read_lock(&tasklist_lock);

> + if (take_lock)

> + read_lock(&tasklist_lock);

> do_each_thread(g, p) {

> if (task_subsys_state(p, subsys_id) == css)

> count ++;

> } while_each_thread(g, p);

> - read_unlock(&tasklist_lock);

> + if (take_lock)

> + read_unlock(&tasklist_lock);

> return count;

Umm, no - please naje two versions with and without the lock. Also

Please fix up the codingstyle, the { belongs onto a line of it's own.
