

---

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.

---