Subject: Re: [PATCH 0/9] Containers (V9): Generic Process Containers
Posted by Srivatsa Vaddagiri on Mon, 30 Apr 2007 17:59:03 GMT
View Forum Message <> Reply to Message

On Mon, Apr 30, 2007 at 10:09:38AM -0700, Paul Menage wrote:
> Paul, is there any reason why we need to do a write_lock() on
> tasklist_lock if we're just trying to block fork, or is it just
> historical accident? Wouldn't it be fine to do a read_lock()?

Good point ..read_lock() will probably suffice in update_nodemask which
means we don't need the patch I sent earlier.

Paul (Jackson),
 This made me see another race in update_nodemask vs fork:

Lets say cpuset CS1 has only one task T1 to begin with.

update_nodemask(CS1) T1 in do_fork()
 CPU0 CPU1
 ========== =

    cpuset_fork();
    mpol_copy();

ntasks = atomic_read(&cs->count);
[ntasks = 2, accounting new born child T2]
cs->mems_allowed = something;
set_cpuset_being_rebound()

write/read_lock(tasklist_lock);

do_each_thread {

 /* Finds only T1 */

 mmarray[] = ..

} while_each_thread();

write/read_unlock(tasklist_lock);

    write_lock(tasklist_lock);

    /* Add T2, child of T1 to tasklist */

write_unlock(tasklist_lock);

for (i = 0; i < n; i++) {

mpol_rebind_mm(..);

}

In this for loop, we migrate only T1's ->mm. T2's->mm isn't migrated AFAICS.

Is that fine?

--
Regards,
vatsa