

---

Subject: Re: [BUG]: Crash with CONFIG\_FAIR\_CGROUP\_SCHED=y  
Posted by [Srivatsa Vaddagiri](#) on Fri, 09 Nov 2007 12:01:16 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Fri, Nov 09, 2007 at 11:59:15AM +0100, Dmitry Adamushko wrote:  
> > - The second problem exposed by this test is that task\_new\_fair()  
> > assumes that parent and child will be part of the same group (which  
> > needn't be as this test shows). As a result, cfs\_rq->curr can be NULL  
> > for the child.  
>  
> Would it be better, logically-wise, to use is\_same\_group() instead?  
> Although, we can't have 2 groups with cfs\_rq->curr != NULL on the same  
> CPU... so if the child belongs to another group, it's cfs\_rq->curr is  
> automatically NULL indeed.

Yeah ..I feel safe with an explicit !curr check, perhaps with a comment like  
below added to explain when curr can be NULL?

---  
kernel/sched\_fair.c | 1 +  
1 files changed, 1 insertion(+)

Index: current/kernel/sched\_fair.c

```
=====
--- current.orig/kernel/sched_fair.c
+++ current/kernel/sched_fair.c
@@ -1022,6 +1022,7 @@ static void task_new_fair(struct rq *rq,
    update_curr(cfs_rq);
    place_entity(cfs_rq, se, 1);

+ /* 'curr' will be NULL if the child belongs to a different group */
+ if (sysctl_sched_child_runs_first && this_cpu == task_cpu(p) &&
+     curr && curr->vruntime < se->vruntime) {
+ /*
```

--  
Regards,  
vatsa

---

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

---