
Subject: [RFC][PATCH 05/16] Use task_pid() to find leader's pid
Posted by [Sukadev Bhattiprolu](#) on Thu, 24 May 2007 01:10:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

Subject: Use task_pid() to find leader's pid

From: Sukadev Bhattiprolu <sukadev@us.ibm.com>

Use task_pid() to get leader's pid since find_pid() cannot be used after detach_pid(). See comments in the code below for more details.

Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>

fs/exec.c | 9 +++++++-
1 file changed, 8 insertions(+), 1 deletion(-)

Index: lx26-21-mm2/fs/exec.c

```
=====
--- lx26-21-mm2.orig/fs/exec.c 2007-05-22 16:59:42.000000000 -0700
+++ lx26-21-mm2/fs/exec.c 2007-05-22 16:59:45.000000000 -0700
@@ -707,10 +707,17 @@ static int de_thread(struct task_struct
 * The old leader becomes a thread of the this thread group.
 * Note: The old leader also uses this pid until release_task
 *      is called. Odd but simple and correct.
+ * Note: With multiple pid namespaces, active pid namespace of
+ * a process is stored in its struct pid. The detach_pid
+ * below frees the struct pid, so we will have no notion
+ * of an active pid namespace until we complete the
+ * subsequent attach_pid(). Which means - calls like
+ * find_pid()/pid_to_nr() return NULL and cannot be used
+ * between the detach_pid() and attach_pid() calls.
 */
detach_pid(tsk, PIDTYPE_PID);
tsk->pid = leader->pid;
- attach_pid(tsk, PIDTYPE_PID, find_pid(tsk->pid));
+ attach_pid(tsk, PIDTYPE_PID, task_pid(leader));
transfer_pid(leader, tsk, PIDTYPE_PGID);
transfer_pid(leader, tsk, PIDTYPE_SID);
list_replace_rcu(&leader->tasks, &tsk->tasks);
```

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