
Subject: [PATCH 1/12] Use find_task_by_vpid in posix timers
Posted by [Pavel Emelianov](#) on Tue, 29 Jan 2008 13:41:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

All the functions that need to lookup a task by pid in posix timers obtain this pid from a user space, and thus this value refers to a task in the same namespace, as the current task lives in.

So the proper behavior is to call find_task_by_vpid() here.

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

```
---
kernel/posix-cpu-timers.c | 8 ++++----
kernel/posix-timers.c     | 2 +-
2 files changed, 5 insertions(+), 5 deletions(-)

diff --git a/kernel/posix-cpu-timers.c b/kernel/posix-cpu-timers.c
index 0b7c82a..2eae91f 100644
--- a/kernel/posix-cpu-timers.c
+++ b/kernel/posix-cpu-timers.c
@@ -20,7 +20,7 @@ static int check_clock(const clockid_t which_clock)
     return 0;

     read_lock(&tasklist_lock);
- p = find_task_by_pid(pid);
+ p = find_task_by_vpid(pid);
     if (!p || !(CPUCLOCK_PERTHREAD(which_clock) ?
         same_thread_group(p, current) : thread_group_leader(p))) {
         error = -EINVAL;
@@ -305,7 +305,7 @@ int posix_cpu_clock_get(const clockid_t which_clock, struct timespec *tp)
    */
    struct task_struct *p;
    rcu_read_lock();
- p = find_task_by_pid(pid);
+ p = find_task_by_vpid(pid);
    if (p) {
        if (CPUCLOCK_PERTHREAD(which_clock)) {
            if (same_thread_group(p, current)) {
@@ -354,7 +354,7 @@ int posix_cpu_timer_create(struct k_itimer *new_timer)
    if (pid == 0) {
        p = current;
    } else {
- p = find_task_by_pid(pid);
+ p = find_task_by_vpid(pid);
        if (p && !same_thread_group(p, current))
            p = NULL;
```

```

}
@@ -362,7 +362,7 @@ int posix_cpu_timer_create(struct k_itimer *new_timer)
    if (pid == 0) {
        p = current->group_leader;
    } else {
-   p = find_task_by_pid(pid);
+   p = find_task_by_vpid(pid);
        if (p && !thread_group_leader(p))
            p = NULL;
    }
diff --git a/kernel/posix-timers.c b/kernel/posix-timers.c
index 80d5620..c75d4e2 100644
--- a/kernel/posix-timers.c
+++ b/kernel/posix-timers.c
@@ -404,7 +404,7 @@ static struct task_struct * good_sigevent(sigevent_t * event)
    struct task_struct *rtn = current->group_leader;

    if ((event->sigev_notify & SIGEV_THREAD_ID) &&
-   (!rtn = find_task_by_pid(event->sigev_notify_thread_id)) ||
+   (!rtn = find_task_by_vpid(event->sigev_notify_thread_id)) ||
        !same_thread_group(rtn, current) ||
        (event->sigev_notify & ~SIGEV_THREAD_ID) != SIGEV_SIGNAL))
        return NULL;
--
1.5.3.4

```
