
Subject: Re: [PATCH] Isolate some explicit usage of task->tgid
Posted by [Oleg Nesterov](#) on Fri, 17 Aug 2007 14:45:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 08/17, Pavel Emelyanov wrote:

>
> Actually the p->tgid == pid has to be changed to has_group_leader_pid(),
> but Oleg pointed out that this is the same and thread_group_leader()
> is more preferable.

No, no, sorry for confusion! I was not clear. I meant that thread_group_leader() is imho better for posix timers, but

> @@ -865,8 +865,8 @@ static int de_thread(struct task_struct
>
> write_lock_irq(&tasklist_lock);
>
> - BUG_ON(leader->tgid != tsk->tgid);
> - BUG_ON(tsk->pid == tsk->tgid);
> + BUG_ON(!same_thread_group(leader, tsk));
> + BUG_ON(thread_group_leader(tsk));

This should be has_group_leader_pid(), BUG_ON() really checks that we didn't confuse ->tgid's.

> /*
> * An exec() starts a new thread group with the
> * TGID of the previous thread group. Rehash the
> diff --git a/fs/proc/base.c b/fs/proc/base.c
> index e3009ab..31e7dfe 100644
> --- a/fs/proc/base.c
> +++ b/fs/proc/base.c
> @@ -2288,7 +2288,7 @@ retry:
> * found doesn't happen to be a thread group leader.
> * As we don't care in the case of readdir.
> */
> - if (!task || !has_group_leader_pid(task))
> + if (!task || !thread_group_leader(task))

And this _must_ be has_group_leader_pid(), this was actually the reason to introduce the "has_group_leader_pid()" helper.

Otherwise I think the patch is fine, and the new helper is really useful.

Oleg.
