
Subject: Re: [PATCH] Check for error returned by kthread_create on creating journal thread

Posted by [Christoph Hellwig](#) on Mon, 16 Apr 2007 11:10:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Mon, Apr 16, 2007 at 03:10:42PM +0400, Pavel Emelianov wrote:

> Christoph Hellwig wrote:

> > On Mon, Apr 16, 2007 at 11:41:14AM +0400, Pavel Emelianov wrote:

> > > If the thread failed to create the subsequent wait_event

> > > will hang forever.

> > >

> > > This is likely to happen if kernel hits max_threads limit.

> > >

> > > Will be critical for virtualization systems that limit the

> > > number of tasks and kernel memory usage within the container.

> >

> > > --- ./fs/jbd/journal.c.jbdthreads 2007-04-16 11:17:36.000000000 +0400

> > > +++ ./fs/jbd/journal.c 2007-04-16 11:30:09.000000000 +0400

> > > @@ -211,10 +211,16 @@ end_loop:

> > > return 0;

> > > }

> > >

> > > -static void journal_start_thread(journal_t *journal)

> > > +static int journal_start_thread(journal_t *journal)

> > > {

> > > - kthread_run(kjournald, journal, "kjournald");

> > > + struct task_struct *t;

> > > +

> > > + t = kthread_run(kjournald, journal, "kjournald");

> > > + if (IS_ERR(t))

> > > + return PTR_ERR(t);

> > > +

> > > wait_event(journal->j_wait_done_commit, journal->j_task != 0);

> >

> > Note that this wait_event should exist at all, and the return

>

> Should NOT you mean?

Umm, yes - of course :)

> > value of kthread_run should be assigned to journal->j_task. Also

> > the code doesn't use the kthread primitives in other places leading

> > to crufty code.

>

> Well, this could be done with a separate patch, I think.
