## Subject: Re: [RFC, PATCH] handle the multi-threaded init's exit() properly Posted by akpm on Tue, 07 Aug 2007 05:18:24 GMT

View Forum Message <> Reply to Message

On Tue, 7 Aug 2007 00:33:26 +0400 Oleg Nesterov <oleg@tv-sign.ru> wrote:

```
> On 08/06, Andrew Morton wrote:
> >
>> On Fri, 3 Aug 2007 01:20:09 +0400 Oleg Nesterov <oleg@tv-sign.ru> wrote:
>>> 2. We are playing games with ->nsproxy->pid_ns. This code is bogus today, and
       it has to be changed anyway when we really support pid namespaces, just
       remove it.
>>>
> > This patch broke
>> pid-namespaces-define-and-use-task active pid ns-wrapper.patch. This hunk:
>> **********
> > *** 908.915 ****
     if (unlikely(!tsk->pid))
      panic("Attempted to kill the idle task!");
>> if (unlikely(tsk == child_reaper(tsk))) {
> > - if (tsk->nsproxy->pid_ns != &init_pid_ns)
      tsk->nsproxy->pid_ns->child_reaper = init_pid_ns.child_reaper;
      else
       panic("Attempted to kill init!");
> >
> >
>> --- 908,916 ----
    if (unlikely(!tsk->pid))
      panic("Attempted to kill the idle task!");
     if (unlikely(tsk == child_reaper(tsk))) {
> >
>> + if (task_active_pid_ns(tsk) != &init_pid_ns)
       task_active_pid_ns(tsk)->child_reaper =
         init_pid_ns.child_reaper;
> > +
      else
       panic("Attempted to kill init!");
> >
> >
> > has no place to live any more, so I just removed it.
> Ah, thanks. I should have done this patch against -mm tree.
> I hope it is OK to drop this chunk of
  pid-namespaces-define-and-use-task_active_pid_ns-wrapper.patch
>
> Because it can't work right now anyway, and Sukadev+Pavel already have
> new patches on top this one which make namespace switch actually work.
```

OK, well I had to make a bit of on-the-fly adjustment to pid-namespaces-rename-child\_reaper-function.patch as well.

The diff-of-the-diff is:

```
@ @ -48,7 +48,7 @ @
diff -puN kernel/exit.c~pid-namespaces-rename-child reaper-function kernel/exit.c
--- a/kernel/exit.c~pid-namespaces-rename-child reaper-function
+++ a/kernel/exit.c
-@@ -694,7 +694,7 @@ forget original parent(struct task struc
+@@ -683,7 +683,7 @@ forget_original_parent(struct task_struc
 do {
  reaper = next_thread(reaper);
  if (reaper == father) {
@@ -57,19 +57,10 @@
  break:
  }
 } while (reaper->exit state);
-@@ -907,7 +907,7 @@ fastcall NORET TYPE void do exit(long co
panic("Aiee, killing interrupt handler!");
if (unlikely(!tsk->pid))
- panic("Attempted to kill the idle task!");
-- if (unlikely(tsk == child_reaper(tsk))) {
-+ if (unlikely(tsk == task_child_reaper(tsk))) {
- if (task_active_pid_ns(tsk) != &init_pid_ns)
 task_active_pid_ns(tsk)->child_reaper =
    init pid ns.child reaper;
diff -puN kernel/signal.c~pid-namespaces-rename-child_reaper-function kernel/signal.c
```

Hopefully people can re-review and retest what's there in next -mm.

Or if that's too much work or too risky, option b) is to drop handle-the-multi-threaded-inits-exit-properly.patch, go back to the 2.6.23-rc1-mm2 versions of pid-namespaces-define-and-use-task\_active\_pid\_ns-wrapper.patch and pid-namespaces-rename-child\_reaper-function.patch and to ask Oleg to cook a 2.6.23-rc1-mm2 version of handle-the-multi-threaded-inits-exit-properly.patch.

The downside of this approach is that handle-the-multi-threaded-inits-exit-properly.patch looks more 2.6.24-ready than all the container stuff (based just on overall impact and speculativeness)

Containers mailing list
Containers@lists.linux-foundation.org

Page 3 of 3 ---- Generated from OpenVZ Forum