Subject: Re: [PATCH 5/5] Move alloc_pid call to copy_process
Posted by Oleg Nesterov on Tue, 17 Jul 2007 13:34:30 GMT
View Forum Message <> Reply to Message

On 07/16, sukadev@us.ibm.com wrote:
>
> Oleg Nesterov [oleg@tv-sign.ru] wrote:
> |
> | Could you please give more details why we need this change?
>
> Well, with multiple pid namespaces, we may need to allocate a new
> 'struct pid_namespace' if the CLONE_NEWPID flag is specified. And
> as a part of initializing this pid_namespace, we need the 'task_struct'
> that will be the reaper of the new pid namespace.
>
> And this task_struct is allocated in copy_process(). So we could
> still alloc_pid() in do_fork(), as we are doing currently and set
> the reaper of the new pid_namespace later in copy_process(). But
> that seemed to complicate error handling and add checks again in
> copy_process() for the CLONE_NEWPID.

OK, thanks.

>
> | Even if we really need this, can't we do these checks in copy_process() ?
>
> We could and I did have a check in copy_process() in one of my earlier
> versions to Containers@ list. We thought it cluttered copy_process() a
> bit.

Yes, but having the "pid == &init_struct_pid" in free_pid() is imho worse,

>   container_exit(p, container_callbacks_done);
>   delayacct_tsk_free(p);
> + free_pid(pid);
> +bad_fork_put_binfmt_module:
> [...snip...]
> @@ -206,6 +206,10 @@ fastcall void free_pid(struct pid *pid)
>   /* We can be called with write_lock_irq(&tasklist_lock) held */
>   unsigned long flags;
>
> + /* check this here to keep copy_process() cleaner */
> + if (unlikely(pid == &init_struct_pid))
> + return;
> +

Wouldn't it better if copy_process()'s error path does

```
 if (pid != &init_struct_pid)
  free_pid(pid);
```

instead? OK, "cleaner" is a matter of taste, but from the perfomance POV
this would be better, even if not noticable.

Oleg.

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