Subject: Re: [PATCH][usercr]: Ghost tasks must be detached
Posted by Oren Laadan on Thu, 10 Feb 2011 14:56:28 GMT
View Forum Message <> Reply to Message

On 02/10/2011 01:17 AM, Sukadev Bhattiprolu wrote:
> Oren Laadan [orenl@cs.columbia.edu] wrote:
> |
> | To address this, initially I thought that we could make ghosts take
> | the tasklist_lock (write) when they change their ->exit_signal.
> |
> | But that's not enough because the parent may already be blocked in
> | wait (so it's too late). Therefore, we also need to have ghosts
> | wake-up their parent through __wake_up_parent().
> |
> | so something like:
> |
> | void ghost_auto_reapable()
> | {
> | write_lock(&tasklist_lock);
> | current->exit_signal = -1;
> | __wake_up_sync_key(current, current->parent);
> | write_unlock(&tasklist_lock);
>
> You meant __wake_up_parent() here I guess.

Yes...

>
> But if we do this in do_ghost_task(), the parent could wakeup, find
> that it still has a live child (this ghost) and go back to sleep before
> the ghost becomes EXIT_DEAD right ?

You are right again...

>
> If so, we would still have the problem ?
>
> i.e we must stop being a chld a of the cinit for it to not wait for us.
> Or we might need to detect that the the pidns is going away, so signalling
> the parent won't cause any harm. But that is racy too :-(

So instead, we can call __wake_up_parent() from exit_checkpoint()
if indeed we are already reaped there:

```
exit_checkpoint()
{
 ...
 if (current->flags & PF_RESTARTING) {
  ...
  /* either zombie or reaped ghost/dead */
  if (current->exit_state = EXIT_DEAD)
   __wake_up_parent(...);   /* probably need lock */
  ...
 }
 ...
}
```

and to avoid userspace misuse, disallow non-thread-group-leader ghosts.

?

Oren.

>
> | }
> |
> | If the parent wasn't at all waiting for us, no harm done...
> |
> | >
> | > So you may ask how did the container-init have a ghost child. That was
> | > due to a bug in usercr :-).
> |
> | You don't need a bug: the ghost flag is used for both ghost and dead
> | tasks (the former used to instantiate specific pids, the latter to
> | make other tasks orphans). So restarting a container that had orphan
> | tasks is guaranteed to do this.
>
> Ah, thats a good point.
>