Subject: Re: [PATCH][usercr]: Ghost tasks must be detached
Posted by Louis Rilling on Tue, 22 Feb 2011 10:28:41 GMT
View Forum Message <> Reply to Message

On 21/02/11 12:40 -0800, Sukadev Bhattiprolu wrote:
> Louis Rilling [Louis.Rilling@kerlabs.com] wrote:
> | > But in 2.6.32 i.e RHEL5, tsk->signal is set to NULL in __exit_signal().
> | > So, I am trying to rule out the following scenario:
> | >
> | > Child (may not be a ghost) Parent
> | > ------------------------ ------
> | > - exit_notify(): is EXIT_DEAD
> | > - release_task():
> | > - drops task_list_lock
> | > - itself proceeds to exit.
> | > - enters release_task()
> | > - sets own->signal = NULL
> | > (in 2.6.32, __exit_signal())
> | >
> | > - enters exit_checkpoint()
> | > - __wake_up_parent()
> | > access parents->signal NULL ptr
> | >
> | > Not sure if holding task_list_lock here is needed or will help.
> |
> | Giving my 2 cents since I've been Cc'ed.
>
> Thanks, appreciate the input :-)
>
> |
> | AFAICS, holding tasklist_lock prevents __exit_signal() from setting
> | parent->signal to NULL in your back. So something like this should be safe:
> |
> | read_lock(&tasklist_lock);
> | if (current->parent->signal)
> | __wake_up_parent(...);
> | read_unlock(&tasklist_lock);
>
> Yes, checking the parent->signal with task_list_lock would work.
>
> |
> | I haven't looked at the context, but of course this also requires that some
> | get_task_struct() on current->parent has been done somewhere else before current
> | has passed __exit_signal().
> |
> | By the way, instead of checking current->parent->signal,
> | current->parent->exit_state would look cleaner to me. current->parent is not
> | supposed to wait on ->wait_chldexit after calling do_exit(), right?

> ^^^^^^^
>
> Hmm, do you mean exit_notify() here ?

Right, I had forgotten zap_pid_ns_processes() ;) My point was just that once ->exit_state is set (for all threads), ->signal->wait_chldexit is not used anymore. But I'm sure that you got it right :)

Thanks,

Louis

>
> If so, yes checking the exit_state is cleaner.
>
> If the parent's exit_state is set, then it can't be waiting for the ghost,
> so no need to wake_up_parent(). If exit state is not set, then it is safe
> to wake_up_parent() (parent->signal would not yet have been cleared for
> instance).
>
> The one case where a parent in do_exit() could still wait for the child is
> the container-init which waits on wait_chldexit in do_exit() ->
> zap_pid_ns_processes() - but even in that case the __wake_up_parent()
> call would be safe.
>
> Sukadev
> _____
> Containers mailing list
> Containers@lists.linux-foundation.org
> https://lists.linux-foundation.org/mailman/listinfo/containe rs

--
Dr Louis Rilling Kerlabs
Skype: louis.rilling Batiment Germanium
Phone: (+33|0) 6 80 89 08 23 80 avenue des Buttes de Coesmes
http://www.kerlabs.com/ 35700 Rennes

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