
Subject: Re: [PATCH] Allow signalling container-init
Posted by [Oleg Nesterov](#) on Thu, 09 Aug 2007 00:02:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 08/08, sukadev@us.ibm.com wrote:

>
> From: Sukadev Bhattiprolu <sukadev@us.ibm.com>
> Subject: [PATCH] Allow signalling container-init
>
> Only the global-init process must be special - any other container-init
> process must be killable to prevent run-away processes in the system.

I think you are right, but....

```
> --- lx26-23-rc1-mm1.orig/kernel/signal.c 2007-08-07 13:52:12.000000000 -0700
> +++ lx26-23-rc1-mm1/kernel/signal.c 2007-08-08 15:09:27.000000000 -0700
> @@ -1861,11 +1861,9 @@ relock:
>     continue;
>
> /*
>  * Init of a pid space gets no signals it doesn't want from
>  * within that pid space. It can of course get signals from
>  * its parent pid space.
> + * Global init gets no signals it doesn't want.
>  */
> - if (current == task_child_reaper(current))
> + if (is_global_init(current->group_leader))
>     continue;
```

...this breaks exec() from /sbin/init. Note that de_thread() kills other sub-threads with SIGKILL. With this patch de_thread() will hang waiting for other threads to die.

I think it is better to not change the current behaviour which is not perfect (buggy), until we actually protect /sbin/init from unwanted signals.

(That said, I am not sure what behaviour is better (worse :), with or without this patch)

Oleg.

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