
Subject: Re: [PATCH] vt: Rework the console spawning variables.

Posted by [ebiederm](#) on Mon, 11 Sep 2006 02:40:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Oleg Nesterov <oleg@tv-sign.ru> writes:

> On 09/10, Eric W. Biederman wrote:

>>

>> Ok. I think I see the where the confusion is. We were looking

>> at different parts of the puzzle. But I we need to resolve this

>> to make certain I didn't do something clever and racy.

>

> Yes, I think we misunderstood each other :)

>

>> As for the rest of your suggestion it would not be hard to be able to

>> follow a struct pid pointer in an rcu safe way, and we do in the pid

>> hash table. In other contexts so far I always have other variables

>> that need to be updated in concert, so there isn't a point in coming

>> up with a lockless implementation. I believe vt_pid is the only

>> case that I have run across where this is a problem and I have

>> at least preliminary patches for every place where signals are

>> sent.

>>

>> Updating this old code is painful.

>

> No, no, we shouldn't change the old code, it is fine.

>

So what happens when:

cpu0: cpu1:

kill_pid(vt_pid,...) fn_SAK()->vc_reset()->put_pid(xchg(&vt_pid, NULL))

Can't kill_pid dereference vt_pid after put_pid is called?

It's a microscopic window, and requires the user to attempt a vt switch
and a sak simultaneously but I think it is there.

Eric

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>
