Subject: Re: [PATCH][usercr]: Ghost tasks must be detached
Posted by Sukadev Bhattiprolu on Wed, 09 Feb 2011 02:09:43 GMT

View Forum Message <> Reply to Message

Oren Laadan [orenl@cs.columbia.edu] wrote:
|
|
| On 02/05/2011 04:40 PM, Sukadev Bhattiprolu wrote:
| > Oren Laadan [orenl@cs.columbia.edu] wrote:
| > | Suka,
| > |
| > | This patch - and the corresponding kernel patch - are wrong
| >
| > Ah, I see that now.
| >
| > But am not sure about the kernel part though. We were getting a crash
| > reliably (with older kernels) because of the ->exit_signal = -1 in
| > do_ghost_task().
|
| Are we still getting it with 2.6.37 ?

I am not currently getting the crash on 2.6.37 - I thought it was due to
the following commit which removed the check for task_detached() in
do_wait_thread().

```
 commit 9cd80bbb07fcd6d4d037fad4297496d3b132ac6b
 Author: Oleg Nesterov <oleg@redhat.com>
 Date:   Thu Dec 17 15:27:15 2009 -0800
```

But if that is true, I need to investigate why Louis Rilling was getting
the crash in Jun 2010 - which he tried to fix here:

http://lkml.org/lkml/2010/6/16/295

Even if we are not currently not getting the crash, I think user-space
actions can result in the container-init being unable to forcibly kill
all its children and exit.

Eg: if ghost tasks are pushed into a child pid namespace (by intentionally
setting ->piddepth in usercr/restart.c), we can have a situation where the
ghost task exits silently, the parent (i.e container-init can be left hanging).

It can be argued that the incorrect changes in usercr code result in the
application hang.

But pid namespace is supposed to guarantee that if a container-init is
terminated, it will take the pid namespace down. But some userspace
actions can result in kill -9 of container-init leaving the container-init

hung forever.

| >
| > One fix I was watching for was Eric Biederman's
| >
| > http://lkml.org/lkml/2010/7/12/213
| >
| > which AFAICT has not been merged yet.
|
| If we need it and it isn't in mainline (any reason why ?) then
| we can just add it to our linux-cr tree, as a preparatory patch.
|
| >
| > Was there another change to 2.6.37 that would prevent the crash ?
|
| I don't know whether *that* crash still happens in 2.6.37 -
| because I still didn't test it with that kernel line back.
| (Actually, I never experienced that crash here even with
| earlier kernels).

Yes, it needed some "accidental" usercr change to expose the crash :-)

(I will try to send a patch to existing usercr and a test case to repro
this problem)