Subject: Re: [PATCH 11/13] Changes to show virtual ids to user
Posted by ebiederm on Thu, 31 May 2007 13:41:22 GMT
View Forum Message <> Reply to Message

Pavel Emelianov <xemul@openvz.org> writes:

> Eric W. Biederman wrote:
>> Pavel Emelianov <xemul@openvz.org> writes:
>>
>>> Pavel Emelianov wrote:
>>>> Cedric Le Goater wrote:
>>>>> Hello !
>>>>>
>>>>>>>>>> The worst case I can see with pid == 0. Is that it would be a bug
>>>>>>>>>> that we can fix later. For other cases it would seem to be a user
>>>>>>>>>> space API thing that we get stuck with for all time.
>>>>>>>> We cannot trust userspace application to expect some pid other than
>>>>>>>> positive. All that we can is either use some always-absent pid or
>>>>>>>> send the signal as SI_KERNEL.
>>>>>>>>
>>>>>>>> Our experience show that making decisions like above causes random
>>>>> <>> applications failures that are hard (or even impossible) to debug.
>>>>>>
>>>>>> Ok. So I guess I see what you are proposing is picking an arbitrary
>>>>>> pid, say pid == 2, and reserving that in all pid namespaces and using
>>>>>> it when we have a pid that does not map to a specific namespace. I'm
>>>>>> fine with that.
>>>>>>
>>>>>> All I care about is that we have a solution, preferably simple,
>>>>>> to the non-mapped pid problem.
>>>>> Pavel, are you against using pid == 0 and setting si_code to SI_KERNEL ?
>>>> I think I am. A quick grep through the code revealed one place where
>>> Sorry. I have misprinted. I meant "I think I am *NOT*". My bad :(
>>>
>>>> this can happen, so I believe application are (have to be) somehow
>>>> prepared to this.
>>
>> Where was this. I'd like to follow your complete line of thinking.
>
> The line concerning why I think that sending a signal from
> SI_KERNEL is good solution?

Let me just restate everything to be certain we are not getting
confused.

The problem was what to do with signals from unmmaped pids.

You have just said pid == 0 with SI_KERNEL seems to work.

The kernel occasionally sends signal that way already.

The primary argument against this in my memory was that we
a user space application might treat the kernel case special
(more trust), so it might be a bad idea.

I believe what you just said was that user space has to be ready
to handle signals from pid == 0 with SI_KERNEL set. Therefore this
should just work. I don't think you have addressed the levels of
trust in user space issue or I might be confused.

Eric
