Subject: Re: namespaces compatibility list
Posted by Cedric Le Goater on Tue, 06 Nov 2007 16:46:09 GMT
View Forum Message <> Reply to Message

Pavel Emelyanov wrote:
> Cedric Le Goater wrote:
>> Pavel Emelyanov wrote:
>>> Cedric Le Goater wrote:
>>>> Pavel Emelyanov wrote:
>>>>> Hi guys!
>>>>>
>>>>> As you might have seen, recently there was some spontaneous
>>>>> discussion about the namespaces-working-together problems.
>>>>>
>>>>> Ted T'so proposed to create some document that describes what
>>>>> problems user may have when he/she creates some new namespace,
>>>>> but keeps others shared. I like this idea, so here's the draft
>>>>> with the problems I currently have in mind and can describe
>>>>> somewhat audibly - the "namespaces compatibility list".
>>>> that compatibility list could be encoded in the way we check
>>>> the clone flags in copy_process() and unshare(). It would
>>>> also be good to have it as a comment somewhere in kernel/fork.c
>>> How can we insure, that a new task will not share the files
>>> with its parent to address the PID namespaces vs VFS namespaces
>>> interaction? There's no way to do it. We can only keep them in
>>> one IPC namespace...
>> ? I'm not sure I understand you.
>
> As far as I understand, you propose the check for the clone flags
> in the copy_process()/sys_unshare() and return -EINVAL for the cases
> we consider to be unsafe. E.g. when a user wants to clone new pid
> namespace, he must clone the ipc namespace as well.

yes.

> But my point is that this check is not enough - user may kill himself
> by cloning a pid namespace and sharing the pids via the filesystem
> (like with the example with futexes) and there's no way to check for
> this situation in the copy_process()/sys_unchare.

right. I think we can address Ulrich concerns first because we have
a solution for it (which looks like unsharing all namespaces at once,
here comes back the container object story :)

> I mean that this list cannot be encoded. But we can warn user, that
> some stuff will stop working if he violates some rules.

and then do that for the futexes, which are a real difficult case.

thanks,

C.