Subject: Re: [PATCH 01/11] SYSCTL: export root and set handling routines
Posted by Stanislav Kinsbursky on Wed, 11 Jan 2012 18:02:16 GMT
View Forum Message <> Reply to Message

>>>>> Especially what drives that desire not to have it have a /proc/<pid>/sys
>>>>> directory that reflects the sysctls for a given process.
>>>>>
>>>>
>>>> This is not so important for me, where to access sysctl's. But I'm worrying
>>>> about backward compatibility. IOW, I'm afraid of changing path
>>>> "/proc/sys/sunprc/*" to "/proc/<pid>/sys/sunrpc". This would break a lot of
>>>> user-space programs.
>>>
>>> The part that keeps it all working is by adding a symlink from /proc/sys
>>> to /proc/self/sys.  That technique has worked well for /proc/net, and I
>>> don't expect there will be any problems with /proc/sys either.  It is
>>> possible but is very rare for the introduction of a symlink in a path
>>> to cause problems.
>>>
>>
>> Probably I don't understand you, but as I see it now, symlink to "/proc/self/"
>> is unacceptable because of the following:
>> 1) will be used current context (any) instead of desired one
> (Using the current context is the desirable outcome for existing tools).
>> 1) if CT has other pid namespace - then we just have broken link.
>
> Assuming the process in question is not in the pid namespace available
> to proc then yes you will indeed have a broken link.  But a broken
> link is only a problem for new applications that are doing something strange.
>

I believe, that container is assuming to work in  it's own network and pid
namespaces.
With your approach, if I'm not mistaken, container's /proc/net and /proc/sys
tunables will be unaccessible from parent environment. Or I'm wrong here?

> I am proposing treating /proc/sys like /proc/net has already been
> treated.  Aka move have the version of /proc/sys that relative to a
> process be visible at: /proc/<pid>/sys, and with a compat symlink
> from /proc/sys ->  /proc/self/sys.
>
> Just like has already been done with /proc/net.
>

1) On one hand it looks logical, that any nested dentries in /proc are tied to
pid namespace. But on the other hand we have a lot of tunables in /proc/net,
/proc/sys, etc. which have nothing with processes or whatever similar.

2) currently /proc processes directories (i.e. /proc/1/, etc) depends on mount
maker context. But /proc/sys and /proc/net doesn't. This looks weird and
despondently, from my pow. What do you think about it?

And what do you think about "conteinerization" of /proc contents in the way like
"sysfs" was done?
Implementing /proc "conteinerization" in this way can give us great flexibility.
For example, /proc/net (and /proc/sys/sunrpc) depends on mount owner net
namespace, /proc/sysvipc depends on mount owner ipc namespace, etc.
And this approach doesn't break backward compatibility as well.

> Semantically this should be easy to understand, and about as backwards
> compatible as it gets.
>
> Eric

--
Best regards,
Stanislav Kinsbursky