Subject: Re: [RFC PATCH 0/5] Resend -v2 - Use procfs to change a syscall behavior
Posted by ebiederm on Fri, 18 Jul 2008 02:49:05 GMT
View Forum Message <> Reply to Message

Matt Helsley <matthltc@us.ibm.com> writes:

> On Thu, 2008-07-17 at 18:42 -0400, Oren Laadan wrote:
>>
>> My question is why build a set of interfaces to export this and that from
>> the kernel to user space ? if a kernel implementation (with minimal user
>> space support) is chosen, then information extraction (and restoration) is
>> straightforward and we don't get ourselves tied until the end of times to
>> API exported to userland.
>
> That still seems like an API exported to userland. It just combines the
> data into one block rather than distributing it amongst a bunch of
> pseudo-filesystems. Does this form of API really free us from always
> supporting it in the future?

A larger granularity reduces the support burden. You don't wind up introducing a bunch of little system calls that you only use for restore. You introduce one that does exactly what you need it to do. Because you know it is only used in checkpoint/restart conditions you can make assumptions about the users and have more freedom.

Yes it would still be a user/kernel interface.

If we abstract it something like binformats are abstracted we may eventually be able to stop including an old format that no one uses anymore.

>
> Userspace is expected to inspect or convert the binary data. How does
> that truly avoid many of the API issues mentioned above? If it's really
> supposed to be a minimal API then the binary should be considered opaque
> and userspace tools which inspect or convert these binaries should be
> considered unreliable hacks at best. Otherwise it seems to me that it
> has most of the familiar problems associated with a kernel/userspace API
> -- including an obligation to support it.

The best precedent we have for something like this today is the core dump. That is a single process and does not do well at tying multiple processes together. Even though you can inspect a core dump there is still a lot of freedom in the implementation that we would not have in a more general API.

As for userspace converting old data to new data. I'm not sold on the

idea yet. It is a good tool to plan on, but I'm not yet convinced
that it is necessary, at least when moving from older to newer kernels.
I expect newer kernels to have state that the older kernels don't know
how to handle, so we would at least need to strip that out.

Eric

_____

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers