

Hey, Eric.

On Thu, Jul 26, 2012 at 12:19:12PM -0700, Eric W. Biederman wrote:

> > No, any attempt to build namespace support into cgroup core code will
> > be nacked with strong prejudice.
>
> The cgroup code was only merged with the understanding that this support
> was simple to add and it would be added. I am sorry that no one had
> the sense to follow up and make certain that promise was not fulfilled.

Good chunk of cgroup is messy and I'm likely to continue to break a lot of whatever promises that have been made. :)

> > Thankfully, procfs is going the FUSE way.
>
> No procfs is not going the FUSE way. Hacks for programs that misuse
> information in procfs is going the FUSE way.

All those were proposed to be solved by "teaching" kernel procfs how to present itself differently.

> The best example is there currently is not a good method for programs
> to figure out how parallel it is productive to be so the programs
> read /proc/cpuinfo and get the count of cpus. Control groups can
> limit you to fewer cpus but those programs have figured that out yet.

Yeah, and you can handle that too nicely with FUSE. More on this later.

> But ultimately fuse for procfs is about the rare case where people
> want to lie to applications, because it is easier to lie to applications
> then to disabuse the applications of their mistaken assumptions.

I don't think so. They are necessary parts of representing a properly scoped environment.

> I have not seen a single suggest that any of the other procfs bits
> can go away.

I think I made that a couple times now. I definitely intend to push things that way. Or, at least, I'll bark as hard as I can against adding more namespace stuff to system pseudo filesystems.

> > and I hope in time we could convert sysfs to a similar mechanism and

> > deprecate the in-kernel support.
>
> I have nothing that even suggests there is a reasonable possibility of
> using fuse to deprecate any of the proc or sysfs support.

Why not? If there's some deficiency in FUSE or notification mechanisms in pseudo FSes, let's fix them.

> > So, no, no, no, no, no, no, no, no, no, no, no, no, no. :P

>

> Bahahahahahahaha! :P

:)

> I sort of wish I had the energy to tackle this. As it is control groups
> hierarchies have very severe usability problems supporting one of their
> core use cases.

>

> We should have our interfaces designed such that it is possible to run
> nested init's without hacks, and the only significant piece left on the
> hacks pile is control groups.

>

> Control group hierarchies are a really strange piece of work whose
> design makes very little sense to me.

cgroupfs is riddled with confused designs but this is not it. The confusion is that namespace should play a major role in the design of system pseudo filesystems and that it can be achieved by playing peekaboo with dentries.

It obfuscates the code for niche use case - which in itself could be acceptable if that's the only / best way to achieve that - while not even being able to serve the said use case properly.

Thanks.

--

tejun
