

Herbert Poetzl <herbert@13thfloor.at> writes:

> On Fri, Dec 08, 2006 at 12:57:49PM -0700, Eric W. Biederman wrote:
>> Herbert Poetzl <herbert@13thfloor.at> writes:
>>
>> >> But, ok, it is not the real point to argue so much imho
>> >> and waste our time instead of doing things.
>
>> > well, IMHO better talk (and think) first, then implement
>> > something ... not the other way round, and then start
>> > fixing up the mess ...
>>
>> Well we need a bit of both.
>
> hmm, are 'we' in a hurry here?

We need to talk about code, and particular patches not just talk.

There are two sides to what we are building.

- The user interface, and semantics.
- The kernel implementation.

For the user interface getting it as close to perfect as we can the first time is extremely important. Because we won't be able to change it.

For the kernel implementation we don't have to be perfect we have to have something that is good enough. We can change the implementation every release if we find better ways of implementing our user space semantics.

> until recently, 'Linux' (mainline) didn't even want
> to hear about OS Level virtualization, now there
> is a rush to quickly get 'something' in, not knowing
> or caring if it is usable at all?
>
> I think there are a lot of 'potential users' for
> this kind of virtualization, and so 'we' can test
> almost all aspects outside of mainline, and once
> we know the stuff works as expected, then we can
> integrate it ...

We should do this as part of the linux kernel community. There is no outside of mainline development. We need to get feedback from

other developers whose code we may effect. This is particularly true of the kernel networking stack.

- > that's something I do not really agree with, stuff
- > integrated into the kernel should be well designed
- > and it should be tested ...

Yes but you can break a problem into reasonable chunks and solve each of those pieces individually.

Eric

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
