
Subject: Re: Which of the virtualization approaches is more suitable for kernel?

Posted by [Herbert Poetzl](#) on Fri, 24 Feb 2006 23:01:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Fri, Feb 24, 2006 at 02:44:42PM -0700, Eric W. Biederman wrote:

> Kirill Korotaev <dev@sw.ru> writes:

>

> > Linus, Andrew,

> >

> > We need your help on what virtualization approach you would accept
> > to mainstream (if any) and where we should go.

> >

> > If to drop VPID virtualization which caused many disputes, we
> > actually have the one virtualization solution, but 2 approaches for
> > it. Which one will go depends on the goals and your approval any
> > way.

>

> My apologies for not replying sooner.

>

> > From the looks of previous replies I think we have some valid
> > commonalities that we can focus on.

>

> Largely we all agree that to applications things should look exactly
> as they do now. Currently we do not agree on management interfaces.

>

> We seem to have much more agreement on everything except pids, so
> discussing some of the other pieces looks worth while.

>

> So I propose we the patches to solve the problem into three categories.

> - General cleanups that simplify or fix problems now, but have

> a major advantage for our work.

> - The kernel internal implementation of the various namespaces

> without an interface to create new ones.

> - The new interfaces for how we create and control containers/namespaces.

proposal accepted on my side

> This should allow the various approach to start sharing code, getting
> progressively closer to each other until we have an implementation we
> can agree is ready to go into Linus's kernel. Plus that will allow us
> to have our technical flame wars without totally stopping progress.

>

> We can start on a broad front, looking at several different things.

> But I suggest the first thing we all look at is SYSVIPC. It is

> currently a clearly recognized namespace in the kernel so the scope is

> well defined. SYSVIPC is just complicated enough to have a non-trivial

> implementation while at the same time being simple enough that we can

> go through the code in exhausting detail. Getting the group dynamics

> working properly.

okay, sounds good ...

> Then we can as a group look at networking, pids, and the other pieces.

>

> But I do think it is important that we take the problem in pieces

> because otherwise it is simply too large to review properly.

definitely

best,
Herbert

> Eric

> -

> To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
> the body of a message to majordomo@vger.kernel.org

> More majordomo info at <http://vger.kernel.org/majordomo-info.html>

> Please read the FAQ at <http://www.tux.org/lkml/>
