Subject: Re: [RFC] [PATCH 0/3] containers: introduction
Posted by serue on Thu, 11 Jan 2007 16:24:52 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> > Quoting Eric W. Biederman (ebiederm@xmission.com):
> >> "Serge E. Hallyn" <serue@us.ibm.com> writes:
> >>
> >> > Or we could go ahead and fully implement it in procfs. As you'd said
> >> > earlier, that really maps best into what we want. Containerfs was
> >> > just much simpler and quicker to implement for demonstrating the semantics.
> >>
> >> Well for what it is worth I just notices that nfs is currently and automounter
> >> that transparently unmounts it's children when you unmount it. I don't think
> >> that is quite enough to split /proc into two but it does have some potential
> >> when it comes to new features.
> >>
> >> Using itty bity purpose built file systems if there is an automounter for them
> >> because much easier for user space.
> >
> > I'm not parsing the last sentence.
> >
> > Are you suggesting that we may be able to stick with a custom fs,
> > using autofs to automount it if the symlink /proc/$$/container is
> > dereferenced while only a kernel mount of /containers exists?
> >
> > I suppose a simpler solution is to not define /proc/$$/container,
> > but rather just let /container in the containerfs symlink to
> > the current process' container. That way you can't reference
> > /containers/container unless containerfs is already mounted under
> > /containers, and we avoid the problem completely.
>
> I am saying:
> autofs is not special. Doing automounting the nfs way
> you can add and remove mounts transparently to the user.

I see, thanks.

We don't want the kernel to know about magic pathname strings,
so as long as we are willing to mount containerfs under a known
location in procfs, this becomes trivial. Otherwise, I guess we
need to talk about convention.

I suppose just not having the kernel-mount, having the symlink,
and making /sys/containers the known location, isn't bad. Then
if /sys/containers isn't mounted and doesn't exist so we can't

automount it, /proc/$$/containers is just a bad link.

> A very good use for this would be to mount/unmount things like
> /proc/sys/fs/binfmt_misc/.
>
> That technique may have an implication for the design of a container
> filesystem.
>
> The result is that if something is more simply implemented as a
> separate filesystem, that is a possibility.

That's what's holding me back here - I'm still not sure whether
to proceed with a separate implementation, proceed with the
current implementation of Paul's containers, or wait for an
update from Paul responding to your feedback.

But both the standalone and paul-based approaches were easy to
implement so I guess it's not a big deal to just proceed with
my own and port to containers if/when appropriate.

-serge

_____

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