Subject: Re: [patch -mm 0/4] mqueue namespace
Posted by serue on Fri, 20 Jun 2008 14:53:25 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> ebiederm@xmission.com (Eric W. Biederman) writes:
>
> > One way to fix that is to add a hidden directory to the mnt namespace.
> > Where magic in kernel filesystems can be mounted. Only visible
> > with a magic openat flag. Then:
> >
> > fd = openat(AT_FDKERN, ".", O_DIRECTORY)
> > fchdir(fd);
> > umount("./mqueue", MNT_DETACH);
> > mount(("none", "./mqueue", "mqueue", 0, NULL);
> >
> > Would unshare the mqueue namespace.
> >
> > Implemented for plan9 this would solve a problem of how do you get
> > access to all of it's special filesystems. As only bind mounts
> > and remote filesystem mounts are available. For linux thinking about
> > it might shake the conversation up a bit.
>
> Thinking about this some more. What is especially attractive if we do
> all namespaces this way is that it solves two lurking problems.
> 1) How do you keep a namespace around without a process in it.
> 2) How do you enter a container.
>
> If we could land the namespaces in the filesystem we could easily
> persist them past the point where a process is present in one if we so
> choose.
>
> Entering a container would be a matter of replacing your current
> namespaces mounts with namespace mounts take from the filesystem.
>
> I expect performance would degrade in practice, but it is tempting
> to implement it and run a benchmark and see if we can measure anything.

The device ns could be a mount of an fs with the devices created in it,
while mknod becomes a symlink from that fs. And once a network
namespace is a filesystem, we can aim for the plan9 NAT solution of
mounting a remote /net onto ours. Neat.

But bye-bye posix?

-serge

_____
Containers mailing list