Subject: Re: [RFC] [PATCH 0/4] uid_ns: introduction
Posted by Herbert Poetzl on Thu, 09 Nov 2006 17:17:01 GMT
View Forum Message <> Reply to Message

On Thu, Nov 09, 2006 at 10:50:09AM -0600, Serge E. Hallyn wrote:
> Quoting Eric W. Biederman (ebiederm@xmission.com):
> > "Serge E. Hallyn" <serue@us.ibm.com> writes:
> >
> > > So from your pov the same objection would apply to tagging vfsmounts,
> > > or not?
> >
> > No. The issue is that the NFS server merges different mounts to the
> > same nfs server into the same superblock.
> >
> > > What is the scenario where the caching is broken? It can't be
> > > multiple clients accessing the same NFS export from the same NFS
> > > service container, since that would just be an erroneous setup,
> > > right?
> >
> > >
> > > > As I recall there are two basic issues.
> > > >
> > > > Putting the default on the mount structure instead of the
> > > > superblock for filesystems that are not uid namespaces aware
> > > > sounded reasonable, and allowed certain classes of sharing
> > > > between namespaces where they agreed on a subset of the uids
> > > > (especially for read-only data).
> > >
> > > yes, that is especially interesting for --bind mounts
> > > when you 'know' that you will dedicate a certain
> > > sub-tree to one context/guest
> >
> > Ok, so you wouldn't object to a patch which tagged vfsmounts?
> >
> > > I guess a NULL vfsmnt->user_ns pointer would mean ignore user_ns and
> > > only apply uid checks (useful for ro bind mount of /usr into multiple
> > > containers).
> >
> > Bind mounts are peculiar. But I think as long as you charged
> > the to the context in which they happen (don't do the bind
> > until after you switch the user_ns. You should be fine.
>
> Presumably container setup would be somewhat like system boot - you'd
> start with a shared / filesystem, unshare user namespace, construct your
> new /, pivot_root, and unmount /old_root, so you end up with all
> vfsmounts accessible from the container having the correct user_ns.

well, once again that is a very narrow view to the

real picture, what about the following cases:

- folks who _share_ certain filesystems between different guests (maybe for cooperation or just readonly to save resource)

- folks who still want a way to access and or andminsitrate the guests (without going through ssh or whatever, e.g. for bulk updates)

- prestructured setups (like build roots) which require pre configured mounts to work ...

best,
Herbert

> -serge
> _____
> Containers mailing list
> Containers@lists.osdl.org
> https://lists.osdl.org/mailman/listinfo/containers

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