Subject: Re: [patch] unprivileged mounts update
Posted by serue on Thu, 26 Apr 2007 16:19:29 GMT
View Forum Message <> Reply to Message

Quoting Miklos Szeredi (miklos@szeredi.hu):
> > Quoting Miklos Szeredi (miklos@szeredi.hu):
> > > > Right, I figure if the normal action is to always do
> > > > mnt->user = current->fsuid, then for the special case we
> > > > pass a uid in someplace. Of course... do we not have a
> > > > place to do that? Would it be a no-no to use 'data' for
> > > > a non-fs-specific arg?
> > >
> > > I guess it would be OK for bind, but not for new- and remounts, where
> > > 'data' is already used.
> > >
> > > Maybe it's best to stay with fsuid after all, and live with having to
> > > restore capabilities. It's not so bad after all, this seems to do the
> > > trick:
> > >
> > > cap_t cap = cap_get_proc();
> > > setfsuid(uid);
> > > cap_set_proc(cap);
> > >
> > > Unfortunately these functions are not in libc, but in a separate
> > > "libcap" library. Ugh.
> >
> > Ok, are you still planning to nix the MS_SETUSER flag, though, as
> > Eric suggested? I think it's cleanest - always set the mnt->user
> > field to current->fsuid, and require CAP_SYS_ADMIN if the
> > mountpoint->mnt->user != current->fsuid.
>
> It would be a nice cleanup, but I think it's unworkable for the
> following reasons:
>
> Up till now mount(2) and umount(2) always required CAP_SYS_ADMIN, and
> we must make sure, that unless there's some explicit action by the
> sysadmin, these rules are still enfoced.
>
> For example, with just a check for mnt->mnt_uid == current->fsuid, a
> fsuid=0 process could umount or submount all the "legacy" mounts even
> without CAP_SYS_ADMIN.
>
> This is a fundamental security problem, with getting rid of MS_SETUSER
> and MNT_USER.
>
> Another, rather unlikely situation is if an existing program sets
> fsuid to non-zero before calling mount, hence unwantingly making that
> mount owned by some user after these patches.

>
> Also adding "user=0" to the options in /proc/mounts would be an
> inteface breakage, that is probably harmless, but people wouldn't like
> it. Special casing the zero uid for this case is more ugly IMO, than
> the problem we are trying to solve.
>
> If we didn't have existing systems to deal with, then of course I'd
> agree with Eric's suggestion.
>
> Miklos

So then as far as you're concerned, the patches which were in -mm will
remain unchanged?

-serge

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers