Subject: Re: Re: [patch 05/10] add "permit user mounts in new namespace" clone flag
Posted by Miklos Szeredi on Tue, 17 Apr 2007 19:43:52 GMT
View Forum Message <> Reply to Message

> > > I'm a bit lost about what is currently done and who advocates for what.
> > >
> > > It seems to me the MNT_ALLOWUSERMNT (or whatever :) flag should be
> > > propagated. In the /share rbind+chroot example, I assume the admin
> > > would start by doing
> > >
> > > mount --bind /share /share
> > > mount --make-slave /share
> > > mount --bind -o allow_user_mounts /share (or whatever)
> > > mount --make-shared /share
> > >
> > > then on login, pam does
> > >
> > > chroot /share/$USER
> > >
> > > or some sort of
> > >
> > > mount --bind /share /home/$USER/root
> > > chroot /home/$USER/root
> > >
> > > or whatever. In any case, the user cannot make user mounts except under
> > > /share, and any cloned namespaces will still allow user mounts.
> >
> > I don't quite understand your method. This is how I think of it:
> >
> > mount --make-rshared /
> > mkdir -p /mnt/ns/$USER
> > mount --rbind / /mnt/ns/$USER
> > mount --make-rslave /mnt/ns/$USER
> > mount --set-flags --recursive -oallowusermnt /mnt/ns/$USER
> > chroot /mnt/ns/$USER
> > su - $USER
> >
> > I did actually try something equivalent (without the fancy mount
> > commands though), and it worked fine. The only "problem" is the
> > proliferation of mounts in /proc/mounts. There was a recently posted
> > patch in AppArmor, that at least hides unreachable mounts from
> > /proc/mounts, so the user wouldn't see all those. But it could still
> > be pretty confusing to the sysadmin.
>
> unbindable mounts were designed to overcome the proliferation problem.
>
> Your steps should be something like this:

>
> mount --make-rshared /
> mkdir -p /mnt/ns
> mount --bind /mnt/ns /mnt/ns
> mount --make-unbindable /mnt/ns
> mkdir -p /mnt/ns/$USER
> mount --rbind / /mnt/ns/$USER
> mount --make-rslave /mnt/ns/$USER
> mount --set-flags --recursive -oallowusermnt /mnt/ns/$USER
> chroot /mnt/ns/$USER
> su - $USER
>
> try this and your proliferation problem will disappear. :-)

Right, this is needed.

My problem wasn't actually this (which would only have hit, if I tried with more than one user), just that the number of mounts in /proc/mounts grows linearly with the number of users.

That can't be helped in such an easy way unfortunately.

> > Propagating some mount flags and not propagating others is
> > inconsistent and confusing, so I wouldn't want that. Currently
> > remount doesn't propagate mount flags, that may be a bug,
>
> For consistency reason, one can propagate all the flags. But
> propagating only those flags that interfere with shared-subtree
> semantics should suffice.

I still don't believe not propagating "allowusermnt" interferes with mount propagation. In my posted patches the mount (including propagations) is allowed based on the "allowusermnt" flag on the parent of the requested mount. The flag is _not_ checked during propagation.

Allowing this and other flags to NOT be propagated just makes it possible to have a set of shared mounts with asymmetric properties, which may actually be desirable.

Miklos