Subject: Re: [patch 1/8] add user mounts to the kernel
Posted by Miklos Szeredi on Sun, 22 Apr 2007 08:05:05 GMT
View Forum Message <> Reply to Message

> >> > + if (mnt->mnt_flags & MNT_USER)
> >> > + seq_printf(m, ",user=%i", mnt->mnt_uid);
> >> How about making the test "if (mnt->mnt_user != &root_user)"
> >
> > We don't want to treat root_user special. That's what capabilities
> > were invented for.
>
> For the print statement? What ever it is minor.

It is a user interface, not a print statement. Your suggested change
would be vetoed by any number of people.

So either we have all mounts having owners, AND have /proc/mounts add
"user=0" to all mounts. While I don't _think_ this would actually
break userspace, it would definitely make people complain.

The other choice is what the current patchset does: is to have
"legacy" mounts without owners, and "new generation" mounts with
owners having "user=UID" in /proc/mounts, regardless of the value of
UID.

> So I want to minimize the changes needed to existing programs.
> Now if all we have to do is specify MS_SETUSER when root a
> user with CAP_SETUID is setting up a mount as a user other
> then himself then I don't much care. If we have to call MS_SETUSER
> as unprivileged users

You don't. Unprivileged mounts _imply_ MS_SETUSER.

> >> > +
> >> > + uid_t mnt_uid;   /* owner of the mount */
> >>
> >> Can we please make this a user struct.   That requires a bit of
> >> reference counting but it has uid namespace benefits as well
> >> as making it easy to implement per user mount rlimits.
> >
> > OK, can you ellaborate, what the uid namespace benifits are?
>
> In the uid namespace the comparison is simpler as are the propagations
> rules. Basically if you use a struct user you will never need to
> care about a uid namespace. If you don't we will have to tear through
> this code another time.

Well, OK. I'll do the user_struct thing then.

Miklos

_____

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