Subject: Re: [PATCH 4/4] The control group itself
Posted by serue on Tue, 15 Jan 2008 14:44:40 GMT
View Forum Message <> Reply to Message

Quoting Pavel Emelyanov (xemul@openvz.org):
> [snip]
>
> > Thanks for working on this, Pavel.
> >
> > My only question with this patch is - so if I create a devs
> > cgroup which only has access to, say /dev/loop0 and /dev/tty3,
> > and someone in that cgroup manages to create a new cgroup, the
> > new cgroup will have all the default permissions again, rather
> > than inherit the permissions from this cgroup, right?
>
> Right. When you create a new cgroup you have an empty perms
> set. Maybe it's worth inheriting the perms from the parent
> container, but I think that empty set is better as you will
> reconfigure it anyway.
>
> [snip]
>
```
> >> +static ssize_t devs_write(struct cgroup *cont, struct cftype *cft,
> >> +  struct file *f, const char __user *ubuf,
> >> +  size_t nbytes, loff_t *pos)
> >> +{
> >> + int err, all, chrdev;
> >> + dev_t dev;
> >> + char buf[64];
> >> + struct devs_cgroup *devs;
> >> + mode_t mode;
```
> >
> > (Of course this will require some privilege, i assume that's a detail
> > you'll add next time around)
>
> Hm... I though that privileges are governed at the cgroup level.... No?

I don't think so... Wouldn't really make sense for the cgroup infrastructure to presume to know what to enforce, and I don't see any checks around the _write functions in cgroup.c, and no capable() calls at all.

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