Subject: Re: Attaching PID 0 to a cgroup
Posted by Dhaval Giani on Tue, 01 Jul 2008 09:47:34 GMT
View Forum Message <> Reply to Message

[put in the wrong alias for containers list correcting it.]

On Tue, Jul 01, 2008 at 03:15:45PM +0530, Dhaval Giani wrote:
> Hi Paul,
>
> Attaching PID 0 to a cgroup caused the current task to be attached to
> the cgroup. Looking at the code,
>
> if (pid) {
> rcu_read_lock();
> tsk = find_task_by_vpid(pid);
> if (!tsk || tsk->flags & PF_EXITING) {
> rcu_read_unlock();
> return -ESRCH;
> }
> get_task_struct(tsk);
> rcu_read_unlock();
>
> if ((current->euid) && (current->euid != tsk->uid)
> && (current->euid != tsk->suid)) {
> put_task_struct(tsk);
> return -EACCES;
> }
> } else {
> tsk = current;
> get_task_struct(tsk);
> }
>
> I was wondering, why this was done. It seems to be unexpected behavior.
> Wouldn't something like the following be a better response? (I've used
> EINVAL, but I can change it to ESRCH if that is better.)
>
> ---
> cgroups: Don't allow PID 0 to be attached to a group
>
> Currently when one trys to attach PID 0 to a cgroup, it attaches
> the current task. That is not expected behavior. It should return
> an error instead.
>
> Signed-off-by: Dhaval Giani <dhaval@linux.vnet.ibm.com>
>
> Index: linux-2.6/kernel/cgroup.c
> ===
> --- linux-2.6.orig/kernel/cgroup.c

> +++ linux-2.6/kernel/cgroup.c
> @@ -1309,8 +1309,7 @@ static int attach_task_by_pid(struct cgr
> return -EACCES;
> }
> } else {
> - tsk = current;
> - get_task_struct(tsk);
> + return -EINVAL;
> }
>
> ret = cgroup_attach_task(cgrp, tsk);
> --
> regards,
> Dhaval

--
regards,
Dhaval

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