Subject: Re: [PATCH -mm] Hook up group scheduler with control groups
Posted by Srivatsa Vaddagiri on Fri, 28 Sep 2007 02:32:11 GMT
View Forum Message <> Reply to Message

On Thu, Sep 27, 2007 at 04:42:41PM -0700, Andrew Morton wrote:
> > @@ -219,6 +225,9 @@ static inline struct task_grp *task_grp(
> >
> > #ifdef CONFIG_FAIR_USER_SCHED
> > tg = p->user->tg;
> > +#elif CONFIG_FAIR_CGROUP_SCHED
> > + tg = container_of(task_subsys_state(p, cpu_cgroup_subsys_id),
> > + struct task_grp, css);
> > #else
> > tg = &init_task_grp;
> > #endif
>
> that's a bit funny-looking. Are CONFIG_FAIR_CGROUP_SCHED and
> CONFIG_FAIR_USER_SCHED mutually exclusive?

Yes. While configuring kernel, user can choose only one of those options
and not both.

> Doesn't seem that way.

Hmm ..why do you say that?

> if
> they're both defined then CONFIG_FAIR_USER_SCHED "wins".
> Anyway, please confirm that this is correct?

They can't both be defined.

> I'll switch that to `#elif defined(CONFIG_FAIR_CGROUP_SCHED)'. We can get
> gcc warnings with `#if CONFIG_FOO', and people should be using `#ifdef
> CONFIG_FOO', so I assume the same applies to #elif.

Thx for fixing it!

--
Regards,
vatsa

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