Subject: Re: [RFC][PATCH] introduce task cgroup (#task restrictioon for prevent fork bomb by cgroup)
Posted by Daisuke Nishimura on Thu, 05 Jun 2008 05:23:20 GMT

View Forum Message <> Reply to Message

Hi.

I think this would be usefull.

On Thu, 05 Jun 2008 13:43:06 +0900, KOSAKI Motohiro <kosaki.motohiro@jp.fujitsu.com> wrote:
> Index: b/kernel/cgroup_task.c
> ==
> --- /dev/null
> +++ b/kernel/cgroup_task.c
> @@ -0,0 +1,185 @@
> +/* cgroup_task.c - #task control group
> + *
> + * Copyright: KOSAKI Motohiro <kosaki.motohiro@jp.fujitsu.com>
> + *
> + * This program is free software; you can redistribute it and/or modify
> + * it under the terms of the GNU General Public License as published by
> + * the Free Software Foundation; either version 2 of the License, or
> + * (at your option) any later version.
> + *
> + * This program is distributed in the hope that it will be useful,
> + * but WITHOUT ANY WARRANTY; without even the implied warranty of
> + * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
> + * GNU General Public License for more details.
> + */
> +
> +#include <linux/res_counter.h>
I don't think it's needed.
Or, are you planning to implement this feature by using res_counter?

> +static int task_cgroup_max_tasks_write(struct cgroup *cgrp,
> +   struct cftype *cftype,
> +   s64 max_tasks)
> +{
> + struct task_cgroup *taskcg;
> +
> + if ((max_tasks > INT_MAX) ||
> +   (max_tasks < INT_MIN))
> + return -EINVAL;
> +
> + taskcg = task_cgroup_from_cgrp(cgrp);
> +
> + spin_lock(&taskcg->lock);
> + if (max_tasks < taskcg->nr_tasks)

> +	return -EBUSY;
need spin_unlock().

Thanks,
Daisuke Nishimura.

_____