
Subject: Re: [PATCH 2/2] CFS CGroup: Report usage
Posted by [Balbir Singh](#) on Tue, 23 Oct 2007 03:17:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul Menage wrote:

```
> Report CPU usage in CFS Cgroup directories
>
> Adds a cpu.usage file to the CFS cgroup that reports CPU usage in
> milliseconds for that cgroup's tasks
>
> This replaces the "example CPU Accounting CGroup subsystem" that
> was merged into mainline last week.
>
> Signed-off-by: Paul Menage <menage@google.com>
>
> ---
> kernel/sched.c | 32 ++++++-----
> 1 file changed, 27 insertions(+), 5 deletions(-)
>
> Index: container-2.6.23-mm1/kernel/sched.c
> =====
> --- container-2.6.23-mm1.orig/kernel/sched.c
> +++ container-2.6.23-mm1/kernel/sched.c
> @@ -7005,15 +7005,37 @@ static u64 cpu_shares_read_uint(struct c
>     return (u64) tg->shares;
> }
>
> -static struct cftype cpu_shares = {
> -    .name = "shares",
> -    .read_uint = cpu_shares_read_uint,
> -    .write_uint = cpu_shares_write_uint,
> +static u64 cpu_usage_read(struct cgroup *cgrp, struct cftype *cft)
> +{
> +    struct task_group *tg = cgroup_tg(cgrp);
> +    int i;
> +    u64 res = 0;
> +    for_each_possible_cpu(i) {
> +        unsigned long flags;
> +        spin_lock_irqsave(&tg->cfs_rq[i]->rq->lock, flags);
> +        res += tg->se[i]->sum_exec_runtime;
> +        spin_unlock_irqrestore(&tg->cfs_rq[i]->rq->lock, flags);
> +    }
> +    /* Convert from ns to ms */
> +    do_div(res, 1000000);
> +    return res;
> +}
> +
```

I think we also need the notion of load, like we have in `cpu_acct.c`
Don't we need to do a `css_get()` on the `cgrp` to ensure that the `cgroup`
does not go away if it's empty and someone does an `rmdir` on it?

```
> +static struct cftype cpu_files[] = {
> +  {
> +    .name = "shares",
> +    .read_uint = cpu_shares_read_uint,
> +    .write_uint = cpu_shares_write_uint,
> +  },
> +  {
> +    .name = "usage",
> +    .read_uint = cpu_usage_read,
> +  },
> };
>
> static int cpu_cgroup_populate(struct cgroup_subsys *ss, struct cgroup
> *cont)
> {
> -  return cgroup_add_file(cont, ss, &cpu_shares);
> +  return cgroup_add_files(cont, ss, cpu_files, ARRAY_SIZE(cpu_files));
> }
>
> struct cgroup_subsys cpu_cgroup_subsys = {
```

--

Warm Regards,
Balbir Singh
Linux Technology Center
IBM, ISTL

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