
Subject: [PATCH v2 2/3] Reuse cgroup's parent pointer
Posted by [Glauber Costa](#) on Mon, 28 Nov 2011 16:45:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

We already have a pointer to the cgroup parent (whose data is more likely to be in the cache than this, anyway), so there is no need to have this one in cpuacct.

This patch makes the underlying cgroup be used instead.

Signed-off-by: Glauber Costa <glommer@parallels.com>
Reviewed-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>
CC: Paul Tuner <pjt@google.com>
CC: Peter Zijlstra <a.p.zijlstra@chello.nl>

kernel/sched/core.c | 15 ++++++++-----
1 files changed, 9 insertions(+), 6 deletions(-)

```
diff --git a/kernel/sched/core.c b/kernel/sched/core.c
index e9e0c19..c36f926 100644
--- a/kernel/sched/core.c
+++ b/kernel/sched/core.c
@@ -7840,7 +7840,6 @@ struct cpuacct {
 /* cpuusage holds pointer to a u64-type object on every cpu */
 u64 __percpu *cpuusage;
 struct percpu_counter cpustat[CPUACCT_STAT_NSTATS];
- struct cpuacct *parent;
};

struct cgroup_subsys cpuacct_subsys;
@@ -7859,6 +7858,13 @@ static inline struct cpuacct *task_ca(struct task_struct *tsk)
 struct cpuacct, css);
}

+static inline struct cpuacct *parent_ca(struct cpuacct *ca)
+{
+ if (!ca || !ca->css.cgroup->parent)
+ return NULL;
+ return cgroup_ca(ca->css.cgroup->parent);
+}
+
 /* create a new cpu accounting group */
static struct cgroup_subsys_state *cpuacct_create(
 struct cgroup_subsys *ss, struct cgroup *cgrp)
@@ -7877,9 +7883,6 @@ static struct cgroup_subsys_state *cpuacct_create(
 if (percpu_counter_init(&ca->cpustat[i], 0))
 goto out_free_counters;
```

```
- if (cgrp->parent)
- ca->parent = cgroup_ca(cgrp->parent);
-
return &ca->css;
```

out_free_counters:

```
@@ -8046,7 +8049,7 @@ void cpuacct_charge(struct task_struct *tsk, u64 cputime)
```

```
ca = task_ca(tsk);
```

```
- for (; ca; ca = ca->parent) {
+ for (; ca; ca = parent_ca(ca)) {
    u64 *cpuusage = per_cpu_ptr(ca->cpuusage, cpu);
    *cpuusage += cputime;
}
```

```
@@ -8088,7 +8091,7 @@ void cpuacct_update_stats(struct task_struct *tsk,
```

```
do {
    __percpu_counter_add(&ca->cpustat[idx], val, batch);
- ca = ca->parent;
+ ca = parent_ca(ca);
} while (ca);
rcu_read_unlock();
}
```

```
--
```

1.7.6.4
