Subject: Re: [RFC][-mm] Simple stats for cpu resource controller v3
Posted by akpm on Fri, 02 May 2008 23:04:34 GMT
View Forum Message <> Reply to Message

On Sat, 3 May 2008 04:17:03 +0530
Balaji Rao <balajirrao@gmail.com> wrote:

> On Saturday 03 May 2008 01:23:04 am Andrew Morton wrote:
> > On Sat, 3 May 2008 01:10:28 +0530
> >
> > Balaji Rao <balajirrao@gmail.com> wrote:
> > > On Friday 02 May 2008 02:30:26 am Andrew Morton wrote:
> > > <snip>
> > >
> > > Hi Andrew,
> > >
> > > Thank you for the review.
> > >
> > > > Did you consider using include/linux/percpu_counter.h?
> > > >
> > > > If so, what was wrong with it?
> > > >
> > > > Because it would be much better to fix per-cpu counters than to invent
> > > > new stuff.
> > >
> > > No, I hadn't consider using the percpu_counters infrastructure. But today
> > > when I tried using it, I got an early exception.I guess its because I
> > > tried calling percpu_counter_init from within sched_init, which I perhaps
> > > shouldn't do, because percpu_counter_init expects cpu hotplug code to be
> > > initialized by then. Right ? Correct me if I'm wrong.
> >
> > I don't see any reason why we cannot run percpu_counter_init() prior to
> > running percpu_counter_startup(). And it is desirable that we be able to
> > start using the percpu-counters quite early.
> >
> > Can you debug it a bit please? It's probably some silly little thing,
> > perhaps fixable by calling percpu_counter_startup() earlier.
> >
> percpu_counter_init uses kmalloc to create percpu counters. This raises an
> early exception as kmem_cache is not initialized that early.

whaa? kmalloc is ready to be used quite early in boot. It's a bit of a
concern that the CPU resource controller is doing stuff before even kmalloc
is ready to go.

What's the call path here? Via cgroup_init_early()? Does it need to run
that early?

> It worked for me if we statically allocate memory for the counters. But its
> not at all a nice thing to do and I don't see another way to make it fit for
> early use.
>
> I'm beginning to run out of ideas! Why not do what I earlier suggested - begin
> collecting statistics once we are able to safely use percpu_counters ? This
> now seems to be the best alternative IMHO.

I'd need to see the code. If we end up doing

```
 if (counters_are_ready)
  increment_counter();
```

all over then place then we need to think harder.

Maybe we need a cgroup_init_late(), which can do memory allocations. If
nothing actually needs to touch the counters before cgroup_init_late() runs
then that might be OK.