Subject: Re: [PATCH, v3 2/2] cgroups: introduce timer slack subsystem
Posted by Kirill A. Shutsemov on Mon, 07 Feb 2011 18:32:24 GMT
View Forum Message <> Reply to Message

On Mon, Feb 07, 2011 at 09:20:40AM -0800, Jacob Pan wrote:
> On Mon, 7 Feb 2011 13:06:03 +0200
> "Kirill A. Shutemov" <kirill@shutemov.name> wrote:
>
> > On Fri, Feb 04, 2011 at 09:27:55AM -0800, Jacob Pan wrote:
> > > On Fri, 4 Feb 2011 15:34:39 +0200
> > > "Kirill A. Shutemov" <kirill@shutemov.name> wrote:
> > > > What's mean "original timer slack" if you are free to move a task
> > > > between a lot of cgroups and process itself free to change it
> > > > anytime?
> > > >
> > >
> > > I need to manage tasks by a management software instead of letting
> > > the task change timer_slack by itself. The goal is to make
> > > management transparent and no modifications to the existing apps.
> > > Therefore, it is desirable to automatically enforce timer_slack
> > > when the apps are in the cgroup while automatically restore it when
> > > it is no longer under cgroup management.
> >
> > Tasks are always under cgroup management. Root cgroup is still cgroup.
> >
> > > So the "original timer slack" can be the default 50us or whatever
> > > value chosen by the task itself. But the app itself should not care
> > > or even be aware of which cgroup it is in.
> > >
> > > So here are two optoins i can think of
> > > 1. add a new variable called cg_timer_slack_ns to struct
> > > task_struct{} cg_timer_slack_ns will be set by cgroup timer_slack
> > > subsystem, then we can retain the original per task value in
> > > timer_slack_ns. timer code will pick max(cg_timer_slack_ns,
> > > timer_slack_ns) if cg_timer_slack_ns is set.
> > >
> > > 2. leave task_struct unchanged, add a current_timer_slack to the
> > > cgroup. timer_slack cgroup does not modify per task timer_slack_ns.
> > > similar to option #1, let timer code pick the timer_slack to use
> > > based on whether the task is in timer_slack cgroup.
> > >
> > > Any thoughts?
> >
> > I think it's over-engineering.
> >
> We do have a real use case that cannot be solved by the current logic,
> I only want to find a solution.
> > What about configuration like this:

> >
> > root_cgroup
> > |--timer_slack.min_slack_ns = 0
> > |--timer_slack.max_slack_ns = ULONG_MAX
> > |--50us
> > | |--timer_slack.min_slack_ns = 50000
> > | |--timer_slack.max_slack_ns = 50000
> > |--500us
> > |--timer_slack.min_slack_ns = 500000
> > |--timer_slack_max_slack_ns = ULONG_MAX
> >
> > If you want a task allow to drive its timer_slack, just leave it in
> > root_cgroup.
> >
> > It you want to drive timer_slack of a task, move it between 50us and
> > 500um based on your policy.
> >
> I surely agree with the dummy root configuration. But when the
> management software moves task among 50us or 500us cgroups, or back to
> dummy root, the timer slack cannot be automatically changed.
>
> consider the following scenarios.
> 1. task_A has timer_slack = ts1 = 3us when it is running in the
> foreground by window manager
> 2. then it is moved to 50us cgroup because it is no longer in the
> foreground, so now ts1 = 50us.
> 3. After a while, the system is running in the low power state, so
> task_A is moved to 500us cgroup, ts1 = 500us. Then the user switch the
> device into normal running state and put task_A in foreground again.
> 4. Management software then moves task_A from 500us cgroup to dummy
> root, but it will not be able to restore the 3us timer slack as needed
> by task_A. Task can surely drive its timer_slack but it breaks the
> cgroup-transparency desired by the management scheme.

Can you use 3us cgroup instead of moving to root cgroup? :)

I understand your use-case, but I can't provide sane interface to
implement this.

> The key is that tasks being managed are not aware of cgroup
> involvement. The management software is the one that moves tasks
> around, but we don't want the management software keeps track of very
> timer slacks value of every tasks.
>
> Thanks,
> Jacob

--

Kirill A. Shutemov

_____

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containe rs