Subject: Re: [ckrm-tech] [RFC] Resource Management - Infrastructure choices
Posted by Pavel Emelianov on Mon, 30 Oct 2006 14:08:03 GMT
View Forum Message <> Reply to Message

[snip]

>
> Consensus/Debated Points
> ----------
>
> Consensus:
>
> - Provide resource control over a group of tasks
> - Support movement of task from one resource group to another
> - Dont support heirarchy for now
> - Support limit (soft and/or hard depending on the resource
> type) in controllers. Guarantee feature could be indirectly
> met thr limits.
>
> Debated:
> - syscall vs configfs interface

1. One of the major configfs ideas is that lifetime of
the objects is completely driven by userspace.
Resource controller shouldn't live as long as user
want. It "may", but not "must"! As you have seen from
our (beancounters) patches beancounters disapeared
as soon as the last reference was dropped. Removing
configfs entries on beancounter's automatic destruction
is possible, but it breaks the logic of configfs.

2. Having configfs as the only interface doesn't alow
people having resource controll facility w/o configfs.
Resource controller must not depend on any "feature".

3. Configfs may be easily implemented later as an additional
interface. I propose the following solution:
- First we make an interface via any common kernel
facility (syscall, ioctl, etc);
- Later we may extend this with configfs. This will
alow one to have configfs interface build as a module.

> - Interaction of resource controllers, containers and cpusets
> - Should we support, for instance, creation of resource
> groups/containers under a cpuset?
> - Should we have different groupings for different resources?

This breaks the idea of groups isolation.

> - Support movement of all threads of a process from one group
> to another atomically?

This is not a critical question. This is something that
has difference in

- move_task_to_container(task);
+ do_each_thread_all(g, p) {
+ if (g->mm == task->mm)
+   move_task_to_container(g);
+ } while_each_thread_all(g, p);

or similar. If we have an infrastructure for accounting and
moving one task_struct into group then solution of how many
task to move in one syscall may be taken, but not the other
way round.

I also add devel@openvz.org to Cc. Please keep it on your replies.
