Subject: Re: [PATCH] cgroup: Convert synchronize_rcu to call_rcu in cgroup_attach_task
Posted by Colin Cross on Tue, 23 Nov 2010 08:58:39 GMT
View Forum Message <> Reply to Message

On Tue, Nov 23, 2010 at 12:14 AM, Li Zefan <lizf@cn.fujitsu.com> wrote:
> 12:06, Colin Cross wrote:
>> The synchronize_rcu call in cgroup_attach_task can be very
>> expensive. All fastpath accesses to task->cgroups that expect
>> task->cgroups not to change already use task_lock() or
>> cgroup_lock() to protect against updates, and, in cgroup.c,
>> only the CGROUP_DEBUG files have RCU read-side critical
>> sections.
>>
>> sched.c uses RCU read-side-critical sections on task->cgroups,
>> but only to ensure that a dereference of task->cgroups does
>> not become invalid, not that it doesn't change.
>>
>
> Other cgroup subsystems also use rcu_read_lock to access task->cgroups,
> for example net_cls cgroup and device cgroup.
I believe the same comment applies as sched.c, I'll update the commit message.

> I don't think the performance of task attaching is so critically
> important that we have to use call_rcu() instead of synchronize_rcu()?
On my desktop, moving a task between cgroups averages 100 ms, and on an Tegra2 SMP ARM platform it takes 20 ms. Moving a task with many threads can take hundreds of milliseconds or more. With this patch it takes 50 microseconds to move one task, a 400x improvement.

>> This patch adds a function put_css_set_rcu, which delays the
>> put until after a grace period has elapsed. This ensures that
>> any RCU read-side critical sections that dereferenced
>> task->cgroups in sched.c have completed before the css_set is
>> deleted. The synchronize_rcu()/put_css_set() combo in
>> cgroup_attach_task() can then be replaced with
>> put_css_set_rcu().
>>
>
>> Also converts the CGROUP_DEBUG files that access
>> current->cgroups to use task_lock(current) instead of
>> rcu_read_lock().
>>
>
> What for? What do we gain from doing this for those debug
> interfaces?
Left over from the previous patch that incorrectly dropped RCU completely. I'll put the rcu_read_locks back.

>> Signed-off-by: Colin Cross <ccross@android.com>
>>
>> ---
>>
>> This version fixes the problems with the previous patch by
>> keeping the use of RCU in cgroup_attach_task, but allowing
>> cgroup_attach_task to return immediately by deferring the
>> final put_css_reg to an rcu callback.
>>
>> include/linux/cgroup.h | 4 +++
>> kernel/cgroup.c | 58 ++++++++++++++++++++++++++++++++++++++----------
>> 2 files changed, 50 insertions(+), 12 deletions(-)
>

_____

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