Subject: Re: Re: Hang with fair cgroup scheduler (reproducer is attached.)
Posted by Steven Rostedt on Sun, 16 Dec 2007 23:17:04 GMT
View Forum Message <> Reply to Message

On Sun, 16 Dec 2007, Dmitry Adamushko wrote:

> Steven,
>
> I guess, there is some analogue of UNLOCKED_CTXSW on -rt
> (to reduce contention for rq->lock).
> So there can be a race schedule() vs. rt_mutex_setprio() or sched_setscheduler()
> for some paths that might explain crashes you have been observing?
>
> I haven't analyzed this case for -rt, so I'm just throwing in the idea in case it can be useful.

I haven't fully analyzed this either, but will look much deeper tomorrow.
I wanted to show you this first just to see if you can easily spot what
went wrong.

I used my logdev logging device
http://rostedt.homelinux.com/logdev

The patch that I used to add the logging is here:
http://rostedt.homelinux.com/rt-bug/debug-logdev.patch

To understand this. lfcnprint(...) is just like printk, but it will print
output the following format:

```
[<timestamp>] cpu:<cpu#> (<current-comm>:<current-pid>) <function>:<line#> <printk-fmt>
```

The tags of lmark() is just

```
[<timestamp>] cpu:<cpu#> (<current-comm>:<current-pid>) <function>:<line#>
```

On context switches, we get

```
>>>> IN LOGDEV SWITCH <<<< cpu: <cpu#>
CPU=<cpu#> [<timestamp>] <prev>:<prev-pid>(<prios>:<task-state>) -->>
<next>:<next-pid>(<prios>)
```

The full dmesg with logdump and error backtrace is here:
http://rostedt.homelinux.com/rt-bug/rt-bug.log

Here's a little snippet of where things went wrong.

```
[94359.652019] cpu:3 (hackbench:1658) pick_next_task_fair:1036 nr_running=1
[94359.652020] cpu:3 (hackbench:1658) pick_next_entity:625 se=ffff810009020800
```

```
[94359.652021] cpu:0 (hackbench:1473) put_prev_entity:631
[94359.652022] cpu:3 (hackbench:1658) pick_next_entity:625 se=ffff81003906b5a8
[94359.652022] cpu:2 (softirq-timer/2:32) put_prev_task_rt:283
[94359.652023] cpu:0 (hackbench:1473) put_prev_entity:631
>>>> IN LOGDEV SWITCH <<<< cpu:3
CPU=3 [94359.652023] hackbench:1658(120:120:120:D) -->> hackbench:1586(120:120:120)
>>>> IN LOGDEV SWITCH <<<< cpu:2
CPU=2 [94359.652024] softirq-timer/2:32(49:115:49:D) -->> softirq-rcu/2:39(49:115:49)
>>>> IN LOGDEV SWITCH <<<< cpu:0
CPU=0 [94359.652025] hackbench:1473(120:120:120:R) -->> hackbench:1591(49:120:120)
[94359.652029] cpu:3 (hackbench:1586) put_prev_entity:631
[94359.652030] cpu:2 (softirq-rcu/2:39) move_tasks:2472
[94359.652030] cpu:3 (hackbench:1586) put_prev_entity:631
[94359.652032] cpu:3 (hackbench:1586) pick_next_task_fair:1036 nr_running=1
[94359.652033] cpu:3 (hackbench:1586) pick_next_entity:625 se=ffff810009020800
[94359.652034] cpu:3 (hackbench:1586) pick_next_entity:625 se=ffff810014c7ab18
[94359.652034] cpu:2 (softirq-rcu/2:39) put_prev_task_rt:283
>>>> IN LOGDEV SWITCH <<<< cpu:3
CPU=3 [94359.652035] hackbench:1586(120:120:120:T) -->> hackbench:1623(120:120:120)
[94359.652036] cpu:2 (softirq-rcu/2:39) pick_next_task_fair:1036 nr_running=1
[94359.652038] cpu:2 (softirq-rcu/2:39) pick_next_entity:625 se=0000000000000000
```

I see that softirq-rcu on cpu 2 started doing a move_tasks, when it got to
the state where nr_running returned 1 and the se from pick_next_entity was
NULL.

This was the run on 2.6.24-rc5-rt1.

I'll look deeper into this on Monday, but if something jumps out at you,
please let me know.

Thanks,

-- Steve