

Re: scheduler (sched_4bsd) questions

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2004-09/0201.html>

From: Stephan Uphoff (ups_at_tree.com)

Date: 09/25/04

To: Julian Elischer <julian@elischer.org>

Date: Sat, 25 Sep 2004 14:00:20 -0400

On Sat, 2004-09-18 at 13:42, Stephan Uphoff wrote:

> On Fri, 2004-09-17 at 21:20, Julian Elischer wrote:

>> Stephan Uphoff wrote:

>>> I am also stomped by the special case of adding a thread X with better

>>> priority than the current thread to the runqueue if they belong to the

>>> same ksegroup. In this case both kg_last_assigned and kg_avail_openings

>>> might be zero and setrunqueue() will not call sched_add().

>>> Because of this it looks like the current thread will neither be

>>> preempted nor will TDF_NEEDRESCHED be set to force rescheduling at the

>>> kernel boundary.

>>> This situation should resolve itself at the next sched_switch – however

>>> this might take a long time. (Especially if essential interrupt threads

>>> are blocked by mutexes held by thread X)

>>>

>>>

>> you are correct. I am not yet preempting a running thread with a lesser

>> priority if they are siblings

>> (unless there is a slot available) This is not because I don't want to

>> do it, but simply because it has not been done yet..

>> we did have NO preemption, so having "some" preemption is still better

>> than where we were.

>> Special case code to check curthread for a preemption could be done but

>> at the moment the decision code for

>> whether to preempt or not is in maybe_preempt() and I don't want to

>> duplicate that. it is on the drawing board though.

>> The other thing is, that even if we should be able to preempt a running

>> thread, there is no guarantee that it is on THIS

>> CPU. It may be on another CPU and that gets nasty in a hurry.

>

> Yes .. this could get nasty.

> This happens when the thread is bound to another cpu or someone changed

> thr_concurrency – otherwise the current thread must be a sibling right ?

>

> Maybe something brutal like:

> if ((curthread->td_ksegrp == kg) &&

> (td->td_priority > curthread->td_priority))

> curthread->td_flags |= TDF_NEEDRESCHED;

freebsd-arch: Re: scheduler (sched_4bsd) questions

>
> *in setrunqueue for*
> *the else case of "if (kg->kg_avail_openings > 0)"*
> *would do the trick (without preemption) for the easy but probably more*
> *common cases?*
>
> *Maybe I can find some time next week to think about a clean*
> *fix. I find it always helpful having a small task in mind while reading*
> *source code.*

I wrote a fix that should cover all cases.
However I would like to test it a little bit before posting the patch.
Is there any multi-threaded kernel torture program that you can
recommend?

Thanks

Stephan

freebsd-arch@freebsd.org mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>
To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"