

Re: [patch] voluntary-preempt-2.6.9-rc1-bk4-Q9

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-09/1370.html>

From: Lee Revell (rlrevell_at_joe-job.com)

Date: 09/04/04

To: Mark_H_Johnson@raytheon.com

Date: Fri, 03 Sep 2004 20:04:17 -0400

On Fri, 2004-09-03 at 11:33, Mark_H_Johnson@raytheon.com wrote:

> Well – THAT was special. Another crash but I may have a clue on both that
> and the "general slow down" in sched.c.
>
> The crash is likely due to a problem with X. I caused it this time when
> I was trying to open / hide Mozilla windows (again preparing to send email
> with a web based client). The last operation that worked was the window
> hide. The one that locked everything up was a click to restore a Mozilla
> window onto the screen. I don't know if this is relevant, but the last
> trace in /var/log/messages was a latency trace caused by X.
>
> Sep 3 09:57:11 dws77 kernel: (X/2382): new 329 us maximum-latency critical
> section.
> Sep 3 09:57:11 dws77 kernel: => started at: <kmap_atomic+0x23/0xe0>
> Sep 3 09:57:11 dws77 kernel: => ended at: <kunmap_atomic+0x7b/0xa0>
>
> I am not sure this is relevant since all the data for it was written
> to disk (my script picked up the latency trace as well). Let me know
> if you want the trace data.
>
> The slow down in sched.c may be due to disk DMA activities.

[...]

> It may be a combination of effects. A question for others doing
> testing (like Lee) – have you been doing any other activity in
> the background when doing your tests? For example, I have found
> that something as simple as
> head -c \$1 /dev/zero >tmpfile [\$1 is a > physical memory size]
> or
> cat tmpfile > /dev/null
> can cause significantly increased latencies in the 2.6 kernels.
>

This is looking more and more like a video driver problem:

"Misbehaving video card drivers are another source of significant delays

in scheduling user code. A number of video cards manufacturers recently began employing a hack to save a PCI bus transaction for each display operation in order to gain a few percentage points on their WinBench [Ziff-Davis 98] Graphics WinMark performance.

The video cards have a command FIFO that is written to via the PCI bus. They also have a status register, read via the PCI bus, which says whether the command FIFO is full or not. The hack is to not check whether the command FIFO is full before attempting to write to it, thus saving a PCI bus read.

The problem with this is that the result of attempting to write to the FIFO when it is full is to stall the CPU waiting on the PCI bus write until a command has been completed and space becomes available to accept the new command. In fact, this not only causes the CPU to stall waiting on the PCI bus, but since the PCI controller chip also controls the ISA bus and mediates interrupts, ISA traffic and interrupt requests are stalled as well. Even the clock interrupts stop.

These video cards will stall the machine, for instance, when the user drags a window. For windows occupying most of a 1024x768 screen on a 333MHz Pentium II with an AccelStar II AGP video board (which is based on the 3D Labs Permedia 2 chip set) this will stall the machine for 25-30ms at a time!"

(from <http://research.microsoft.com/~mbj/papers/tr-98-29.html>)

Ingo, would the above situation indeed produce these symptoms?

I had this exact problem with my Via Unichrome chipset (an open source driver!), reported it to the maintainers, and it turned out this was exactly what the driver was doing (it's fixed now). The above text is from 1998, the problem in the via driver was discovered last week. So I would actually expect this behavior to be the norm, unless someone has fixed that driver.

The easiest way to eliminate this possibility is to disable DRI and set 'Option "NoAccel"' in your X config. Do you get the same mysterious latencies with this setting?

What video hardware are you using?

Lee

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>