Subject: Re: i2o hardware hangs (ASR-2010S)
Posted by Markus Lidel on Mon, 07 Aug 2006 14:42:20 GMT
View Forum Message <> Reply to Message

Hello,

Vasily Averin wrote:
> thank you for your assistance.
> Salyzyn, Mark wrote:
>> Markus, when the commands time out, do you perform a reset iop sequence?
>> I thought you added the BlinkLED code detection that is in the dpt_i2o
>> driver, if not, we should make sure it is there so that we get a report
>> in the console and an accompanying reset. Vasily, you console log did
>> not report anything at the time of failure, I would have expected some
>> timeout reports.
> Unfortunately console logs does not have any errors or timeout reports.
> If you wish, I can send you console logs directly.
> However as far as I understand i2o layer does not have any sort of timeout/error
> handlers (I hope Markus correct me if I'm err), and it would be great if this
> feature will be appear in the future.

You're right, there is no timeout/error handling at all, because the I2O spec says there must be a response to every message sent to the controller. The controller must implement an timeout feature itself (for example if a disk disappears during an operation, the controller take care of it and informs the OS that something is wrong). But if this really happens there should be at least a error message.

>> If it will help, Vasily, contact me for the latest dpt_i2o driver as
>> that is the driver I am most familiar with; it may be of interest to
>> determine if the problem duplicates with the dpt_i2o driver. Keep in
>> mind that the i2o driver is a block driver, dpt_i2o is a scsi driver.
>
> Unfortunately we do not know how we can reproduce this issue. Currently it
> occurs on the production nodes only and customers are very against of any
> experiments on these nodes.
> Therefore it is not to easy to switch from i2o layer to your dpt_i2o driver.
> Currently we have not dpt_i2o driver in our kernels. The most important reasons are:
> - this driver did have some problems on 64-bit kernels (but it is resolved
> already, I'm I right?).
> - it is not included into 2.6-based Red Hat distributiuons.
> - it did not worked when I've tried to compile it into kernel.
> - when I've tried to build it as module, I've discovered that it conflicts with
> i2o drivers: initscripts on the some distributions (FC4?) have tried to load
> both of these modules (one from initrd, second -- when detects according PCIID)
> and it hangs the node. I've not found any working combination and therefore
> we've decided to not include dpt_i2o driver into our 2.6 kernels.
> However, Mark, I'm ready to check your new driver on our internal testnodes, and

> if last issue (modules conflicts) is fixed I'll try to include your driver into
> our kernels.

With the latest version the module conflicts should be fixed.

>>> -----Original Message-----
>>> From: linux-scsi-owner@vger.kernel.org
>>> [mailto:linux-scsi-owner@vger.kernel.org] On Behalf Of Vasily Averin
>>> Sent: Friday, August 04, 2006 7:50 AM
>>> To: linux-scsi@vger.kernel.org; Markus Lidel
>>> Cc: devel@openvz.org
>>> Subject: i2o hardware hangs (ASR-2010S)
>>>
>>>
>>> Hello Markus,
>>>
>>> We experience problems with I2O hardware on 2.6 kernels,
>>> probably this can help
>>> you or maybe you even know the answer. Can you please, take a look?
>>>
>>> After migration to 2.6 kernels our customers began to claim
>>> that i2o-based
>>> nodes hang. We have investigated these claims and discovered
>>> that i2o disks on
>>> theses nodes stopped the processing of any IO requests.
>>> Please, note, it is not
>>> a single issue, it happens from time to time.
>>>
>>> Our kernel-space watchdog module has produced the following
>>> output to serial console
>>>
>>> Jul 31 07:38:37
>>> (80,0) i2o/hda r(77135616 1632632476 15538880) w(69903626
>>> 1034743472 407332291)
>>> Jul 31 07:39:38
>>> (80,0) i2o/hda r(77148190 1633252850 15543968) w(69906364
>>> 1034764548 407338084)
>>> (80,0) i2o/hda r(77157038 1633672916 15546672) w(69912375
>>> 1034808048 407351490)
>>> (80,0) i2o/hda r(77169933 1634285356 15550897) w(69916317
>>> 1034845588 407364374)
>>> (80,0) i2o/hda r(77178290 1634941276 15555039) w(69919031
>>> 1034865212 407369386)
>>> (80,0) i2o/hda r(77192170 1635427776 15559925) w(69922676
>>> 1034892406 407377617)
>>> (80,0) i2o/hda r(77216478 1635774384 15570783) w(69927294
>>> 1034921708 407385382)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966

>>> 1034928376 407387163)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928378 407387163)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928384 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928384 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928384 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928386 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928390 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928390 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928390 407387164)
>>> (80,0) i2o/hda r(77221642 1635925752 15572389) w(69927966
>>> 1034928390 407387164)
>>>
>>> where r(reads, read_sectors, read_merges) w(writes,
>>> write_sectors, write_merges)
>>>
>>> Magic keys works, according to showProcess processors are in
>>> idle, ShowTraces
>>> shows a few thousand processes in D-state, but we can not
>>> find any deadlocks, it
>>> looks like the processes waits until I/O finished.
>>> Unfortunately i2o layer has
>>> no any error handlers and there is no any chance that the
>>> node will return
>> >from this coma.
>>> Described incident has occurred after ~2 weeks uptime. It was
>>> Supermicro X5DP8
>>> motherboard /8Gb memory /Adaptec ASR-2010S I2O Zero Channel. Kernel
>>> 2.6.8-022stab078.9-enterprise, sources/configs are accessible
>>> on openvz.org.
>>>
>>> In the bootlogs I've found mtrr message. As far as I know you
>>> have fixed this
>>> issue, however I'm not sure that it can leads to described hang.
>>>
>>> I2O Core - (C) Copyright 1999 Red Hat Software
>>> i2o: max_drivers=4
>>> i2o: Checking for PCI I2O controllers...
>>> ACPI: PCI interrupt 0000:06:01.0[A] -> GSI 72 (level, low) -> IRQ 72
>>> i2o: I2O controller found on bus 6 at 8.
>>> i2o: PCI I2O controller

>>>   BAR0 at 0xF8400000 size=1048576
>>>   BAR1 at 0xFB000000 size=16777216
>>> mtrr: type mismatch for fb000000,1000000 old: uncachable new:
>>> write-combining
>>> i2o: could not enable write combining MTRR
>>> iop0: Installed at IRQ 72
>>> iop0: Activating I2O controller...
>>> iop0: This may take a few minutes if there are many devices
>>> iop0: HRT has 1 entries of 16 bytes each.
>>> Adapter 00000012: TID 0000:[HPC*]:PCI 1: Bus 1 Device 22 Function 0
>>> iop0: Controller added
>>> I2O Block Storage OSM v0.9
>>>   (c) Copyright 1999-2001 Red Hat Software.
>>> block-osm: registered device at major 80
>>> block-osm: New device detected (TID: 211)
>>> Using anticipatory io scheduler
>>> i2o/hda: i2o/hda1 i2o/hda2 < i2o/hda5 i2o/hda6 >
>>>
>>> # cat /proc/mtrr
>>> reg00: base=0xf8000000 (3968MB), size= 128MB: uncachable, count=1
>>> reg01: base=0x00000000 (0MB), size=8192MB: write-back, count=1
>>> reg02: base=0x200000000 (8192MB), size= 128MB: write-back, count=1
>>> reg03: base=0xf7f80000 (3967MB), size= 512KB: uncachable, count=1
>>>
>>> I would repeat, it is not a single fault, we have received
>>> similar claims once
>>> and again. For some time we believed that it was due some
>>> hardware faults,
>>> however some doubts are cast upon it. The same nodes worked
>>> well long time ago
>>> without any troubles under 2.4-based kernels with dpt_i2o
>>> driver and we have not
>>> observed any of i2o hardware troubles so frequently.
>>>
>>> Is it possible that our kernel (based on 2.6.8.1 mainstream)
>>> have some bugs in
>>> i2o drivers? However we're using driver sources taken from
>>> RHEL4U2 kernel, and I
>>> cannot find any similar claims from RHEL4 customers.
>>>
>>> Is it possible than we have some other related kernels bugs?
>>> In this case why we
>>> have such kind of issues only on i2o-based nodes?
>>>
>>> Could you please give me some hints which allow me to
>>> continue investigation of
>>> this issue. If you have any suggestions I'll check them next time.
>>>

>>> Thank you,
>>> Vasily Averin
>>>
>>> SWsoft Virtuozzo/OpenVZ Linux kernel team

Best regards,

Markus Lidel

Markus Lidel (Senior IT Consultant)

Shadow Connect GmbH
Carl-Reisch-Weg 12
D-86381 Krumbach
Germany

Phone: +49 82 82/99 51-0
Fax: +49 82 82/99 51-11

E-Mail: Markus.Lidel@shadowconnect.com
URL: http://www.shadowconnect.com