
Subject: Re: i2o hardware hangs (ASR-2010S)
Posted by [Markus Lidel](#) on Tue, 08 Aug 2006 21:55:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

Salyzyn, Mark wrote:

> I had sent you the driver source in a previous email, I am sending it
> again. Please keep me in the loop since latest model kernels (we have
> customers that confirm 2.6.16) may require changes in the driver to
> compile.
> Since the kernel.org policy is to focus on the i2o driver being beefed
> up, no patches or changes are accepted for the dpt_i2o driver into the
> kernel. Sad that we had just finished a stint beefing up the dpt_i2o
> driver just before that decision was made ...
> The comments about error recovery were meant as a starting point, it
> looks like Markus will have the final say.

Hmmm, personally i would only add error recovery if the behaviour
couldn't be solved otherway (in this case the problem is solved already
in recent kernels), because the controller should already handle it
(regarding to the I2O spec). But if it is wanted i would add it.

> As for the timeouts, I referred to DASD (Disk) targets. 3 minute for
> RAID devices in a rolling timeout is used to deal with situations that
> require a complete spin up of all component drives, or to deal with
> worst case error recovery scenarios. Individual DASD targets, on the
> other hand, should report back within 30 seconds for I/O. None DASD
> targets are all direct, and thus should respect any timeouts set by the
> system (if any).

>> -----Original Message-----

>> From: Vasily Averin [mailto:vvs@sw.ru]

>> Sent: Tuesday, August 08, 2006 5:48 AM

>> To: Salyzyn, Mark

>> Cc: Markus Lidel; devel@openvz.org

>> Subject: Re: i2o hardware hangs (ASR-2010S)

>> Salyzyn, Mark wrote:

>>> Vasily, it will necessarily be up to you as to whether you switch to
>>> dpt_i2o to get the hardening you require today, or work out
>> a deal with

>>> Markus to add timeout/reset functionality to the i2o driver.

>> Of course, you are right. Currently our customers have bad 2
>> alternatives:

>> - be tolerate to these hangs

>> - if they can't bear it -- replace i2o hardware

>> Therefore first at all I'm going to add third possible
>> alternative, dpt_i2o driver.

>> Mark, could you please send me latest version of your driver

>> directly? Or can I
>> probably take it from mainstream?
>> The next task is help Markus in i2o error/reset handler
>> implementation.

Hmmm, the 2.6.8 kernel is very old in terms of my work. The changes made to this kernel were just to get something working at all. In more recent kernels (expect 2.6.16, which is broken) it should work fine without the hangup (in the early versions of the kernel the messages transferred to the controller was too large, which led to the hangup you reported). I would suggest at least 2.6.13 if possible.

>>> My recommendations for the i2o driver reset procedure is to use a
>>> rolling timeout, every new command completion resets the
>> global timer.
>>> This will allow starved or long commands to process. Once
>> the timer hits
>>> 3 minutes for RAID (Block or SCSI) targets that have multiple
>>> inheritances, 30 seconds for SCSI DASD targets, or some
>> insmod tunable,
>>> it resets the adapter. I recommend that when we hit ten
>> seconds, or some
>>> insmod tunable, that we call a card specific health check
>> routine. I do
>>> not recommend health check polling because we have noticed
>> a reduction
>>> in Adapter performance in some systems and generic i2o cards would
>>> require a command to check, so that is why I tie it to the
>> ten seconds
>>> past last completion. For the DPT/Adaptec series of
>> adapters, it checks
>>> the BlinkLED status (code fragment in dpt_i2o driver at
>>> adpt_read_blink_led), and if set, immediately record the
>> fact and resets
>>> the adapter. For cards other than the DPT/Adaptec series, I
>> recommend a
>>> short timeout Get Status request to see if the Firmware is in a run
>>> state and is responsive to this simple command. The reset
>> code will need
>>> to retry all commands itself, I do not believe the block
>> system has an
>>> error status that can be used for it to retry the commands.
>> If the Reset
>>> loop in the reset adapter code is unresponsive, then the
>> known targets
>>> need to be placed offline.
>> Sorry, I do not have your big experience in scsi and do not
>> know nothing in i2o.

>> However are you sure than 3 min is enough for timeout? As far
>> as I know some
>> scsi commands (for example rewind on tapes) can last during a
>> very long time.
>>
>> Also I have some other questions but currently I'm not fell
>> that I'm ready for
>> this discussion.

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>
