Subject: megaraid_mbox: garbage in file
Posted by vaverin on Thu, 04 May 2006 18:46:04 GMT
View Forum Message <> Reply to Message

Hello all,

I've investigated customers claim on the unstable work of their node and found a strange effect: reading from some files leads to the "attempt to access beyond end of device" messages.

I've checked filesystem, memory on the node, motherboard BIOS version, but it does not help and issue still has been reproduced by simple file reading.

Reproducer is simple:

```
echo 0xffffffff >/proc/sys/dev/scsi/logging_level ;
cat /vz/private/101/root/etc/ld.so.cache >/tmp/ttt  ;
echo 0 >/proc/sys/dev/scsi/logging
```

It leads to the following messages in dmesg

```
sd_init_command: disk=sda, block=871769260, count=26
sda : block=871769260
sda : reading 26/26 512 byte blocks.
scsi_add_timer: scmd: f79ed980, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf79ed980                 sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
buffer = 0xf7cfb540, bufflen = 13312, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f79ed980, rtn: 1
sd 0:1:0:0: done 0xf79ed980 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
26 sectors total, 13312 bytes done.
use_sg is 4
attempt to access beyond end of device
sda6: rw=0, want=1044134458, limit=951401367
Buffer I/O error on device sda6, logical block 522067228
attempt to access beyond end of device
sda6: rw=0, want=1178878530, limit=951401367
Buffer I/O error on device sda6, logical block 589439264
...
```

As far as I see first read operation has finished without errors, but when we read the rest of file we get an access to beyond end of device.

Originally it was found on Virtuozzo kernels (2.6.8.1-based x86 32-bit), reproduced on RHEL4 kernels 2.6.9-22.EL and 2.6.9-34.EL, on FC5 (2.6.16-1.2096_FC5) and on vanilla 2.6.16 kernels.

However, when I first read these blocks by using dd with bs=512 or 1024 it works without any troubles. Then I can cat this file, copy it, map it and so on -- and get correct content without any errors. Moreover, this issue may be workarounded by memory limitation: it helps to use mem=4G in kernel commandline or kernels without PAE support.

I've noticed that we attempt to access to the blocks with a strange numbers:

522067228 = 0x1f1e1d1c
589439264 = 0x23222120 and so on.

Then I've found that I've read strange garbage from file:

```
# hexdump /tmp/ttt
0000000 0100 0302 0504 0706 0908 0b0a 0d0c 0f0e
0000010 1110 1312 1514 1716 1918 1b1a 1d1c 1f1e
0000020 2120 2322 2524 2726 2928 2b2a 2d2c 2f2e
0000030 3130 3332 3534 3736 3938 3b3a 3d3c 3f3e
0000040 4140 4342 4544 4746 4948 4b4a 4d4c 4f4e
0000050 5150 5352 5554 5756 5958 5b5a 5d5c 5f5e
0000060 6160 6362 6564 6766 6968 6b6a 6d6c 6f6e
0000070 7170 7372 7574 7776 7978 7b7a 7d7c 7f7e
0000080 0100 0302 0504 0706 0908 0b0a 0d0c 0f0e
0000090 1110 1312 1514 1716 1918 1b1a 1d1c 1f1e
00000a0 2120 2322 2524 2726 2928 2b2a 2d2c 2f2e
...
00000f0 7170 7372 7574 7776 7978 7b7a 7d7c 7f7e
0000100 0100 0302 0504 0706 0908 0b0a 0d0c 0f0e
...
```

Then I've discovered that "access beyond end of device" occurs due reading of the same garbage from the 13-th (Indirect) block of the file.

I've tried to understand where we got this garbage and found that it is present in the data buffers beginning at megaraid_mbox driver functions.

Could somebody explain me what is the strange garbage: repeated 0...127?
Seokmann, Atul, could you please tell me if it is a known issue?
James, from my point of view it is not looks like a driver bug, but probably I'm wrong?

I suppose it is MegaRAID SATA 150-4 firmware issue. I've seen similar firmware fixes for MegaRAID SATA 300 controllers ("Support PAE mode fixed" and "Fixed the operating systems using more than 4 gig of memory"). Is it probably the same issues are present in SATA 150-4 firmware? Or may be I use broken controller?

Hardware Environment:
Tyan B2881
2 x Opteron 246
8G RAM
LSI MegaRAID SATA 150-4
/vz partition formatted as ext3 with 1Kb blocksize

```
megaraid cmm: 2.20.2.6 (Release Date: Mon Mar 7 00:01:03 EST 2005)
megaraid: 2.20.4.7 (Release Date: Mon Nov 14 12:27:22 EST 2005)
megaraid: probe new device 0x1000:0x1960:0x1000:0x4523: bus 1:slot 4:func 0
ACPI: PCI Interrupt 0000:01:04.0[A] -> GSI 29 (level, low) -> IRQ 16
megaraid: fw version:[713N] bios version:[G119]
scsi0 : LSI Logic MegaRAID driver
scsi[0]: scanning scsi channel 0 [Phy 0] for non-raid devices
scsi[0]: scanning scsi channel 1 [virtual] for logical drives
  Vendor: MegaRAID  Model: LD 0 RAID1  476G  Rev: 713N
  Type:   Direct-Access                      ANSI SCSI revision: 02
```

Also I would note that from my point of view this issue looks similar to
http://bugzilla.kernel.org/show_bug.cgi?id=6052

It seems for me both of our cases may have the same cause.

Thank you,
 Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: Re: megaraid_mbox: garbage in file

Posted by James Bottomley on Thu, 04 May 2006 22:59:27 GMT

View Forum Message <> Reply to Message

On Thu, 2006-05-04 at 22:48 +0400, Vasily Averin wrote:

```
> attempt to access beyond end of device
> sda6: rw=0, want=1044134458, limit=951401367
> Buffer I/O error on device sda6, logical block 522067228
```

That's not a SCSI error. It's coming from the block layer and it means that the filesystem tried to access beyond the end of the listed partition.   Why that happened is anyone's guess. I suspect the actual filesystem is corrupt somehow, but how it came to be, I don't know.

James

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 05 May 2006 05:34:39 GMT

View Forum Message <> Reply to Message

James Bottomley wrote:
> On Thu, 2006-05-04 at 22:48 +0400, Vasily Averin wrote:
>>attempt to access beyond end of device
>>sda6: rw=0, want=1044134458, limit=951401367
>>Buffer I/O error on device sda6, logical block 522067228
>
> That's not a SCSI error. It's coming from the block layer and it means
> that the filesystem tried to access beyond the end of the listed
> partition.   Why that happened is anyone's guess. I suspect the actual
> filesystem is corrupt somehow, but how it came to be, I don't know.

James,

The issue is that the correctly finished scsi read command return me garbage
(repeated 0 ...127 -- see hexdump in my first letter) instead correct file content.
"attempt to access beyond end of device" messages occurs due the same garbage
readed from the Indirect block. I found this garbage present in data buffers
beginning at megaraid driver functions.

I would note that if I read the same file by using dd with bs=1024 or bs=512 --
I get correct file content.

When I use kernel with 4Gb memory limit -- the same cat command return me
correct file content too, without any garbage.

Question is what it is the strange garbage? Have you seen it earlier?
Is it possible that it is some driver-related issue or it is broken hardware?
And why I can workaround this issue by using only 4Gb memory?

Thank you,
Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 05 May 2006 09:18:57 GMT

View Forum Message <> Reply to Message

Small update:

When I use
cat /vz/private/101/root/etc/ld.so.cache >/tmp/ttt
I've get "access beyond end of device" and garbage in buffers

Then I create the same scsi read command by using sgp_dd utils:
sgp_dd count=26 if=/dev/sg0 skip=871769260 of=/tmp/ttt.sgp
and get correct file content without any errors.

The only difference that I see is use_sg=3 for cat and use_sg=1 for dd.

dmesg with scsi debugs and output files are attached.

Node will be accessible for some time and I can perform some experiments. If somebody wants I can request the customer about access on the node.

Thank you,
Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Vasily Averin wrote:
> James Bottomley wrote:
>>On Thu, 2006-05-04 at 22:48 +0400, Vasily Averin wrote:
>>>attempt to access beyond end of device
>>>sda6: rw=0, want=1044134458, limit=951401367
>>>Buffer I/O error on device sda6, logical block 522067228
>>That's not a SCSI error. It's coming from the block layer and it means
>>that the filesystem tried to access beyond the end of the listed
>>partition.   Why that happened is anyone's guess. I suspect the actual
>>filesystem is corrupt somehow, but how it came to be, I don't know.
>
> James,
>
> The issue is that the correctly finished scsi read command return me garbage
> (repeated 0 ...127 -- see hexdump in my first letter) instead correct file content.
> "attempt to access beyond end of device" messages occurs due the same garbage
> readed from the Indirect block. I found this garbage present in data buffers
> beginning at megaraid driver functions.
>
> I would note that if I read the same file by using dd with bs=1024 or bs=512 --
> I get correct file content.
>
> When I use kernel with 4Gb memory limit -- the same cat command return me
> correct file content too, without any garbage.
>
> Question is what it is the strange garbage? Have you seen it earlier?
> Is it possible that it is some driver-related issue or it is broken hardware?
> And why I can workaround this issue by using only 4Gb memory?
>
> Thank you,
> Vasily Averin

>
> SWsoft Virtuozzo/OpenVZ Linux kernel team
>

```
Linux version 2.6.16 (vvs@dhcp0-157) (gcc version 3.3.5 20050117 (prerelease) (SUSE Linux))
#1 SMP Thu May 4 17:49:16 MSD 2006
BIOS-provided physical RAM map:
 BIOS-e820: 0000000000000000 - 000000000009fc00 (usable)
 BIOS-e820: 000000000009fc00 - 00000000000a0000 (reserved)
 BIOS-e820: 00000000000e0000 - 0000000000100000 (reserved)
 BIOS-e820: 0000000000100000 - 00000000fbff0000 (usable)
 BIOS-e820: 00000000fbff0000 - 00000000fbfff000 (ACPI data)
 BIOS-e820: 00000000fbfff000 - 00000000fc000000 (ACPI NVS)
 BIOS-e820: 00000000ff780000 - 0000000100000000 (reserved)
 BIOS-e820: 0000000100000000 - 0000000200000000 (usable)
7296MB HIGHMEM available.
896MB LOWMEM available.
found SMP MP-table at 000ff780
NX (Execute Disable) protection: active
On node 0 totalpages: 2097152
  DMA zone: 4096 pages, LIFO batch:0
  DMA32 zone: 0 pages, LIFO batch:0
  Normal zone: 225280 pages, LIFO batch:31
  HighMem zone: 1867776 pages, LIFO batch:31
DMI 2.3 present.
ACPI: RSDP (v002 ACPIAM                                ) @ 0x000f6dd0
ACPI: XSDT (v001 A M I  OEMXSDT  0x12000527 MSFT 0x00000097) @ 0xfbff0100
ACPI: FADT (v001 A M I  OEMFACP  0x12000527 MSFT 0x00000097) @ 0xfbff0281
ACPI: MADT (v001 A M I  OEMAPIC  0x12000527 MSFT 0x00000097) @ 0xfbff0380
ACPI: OEMB (v001 A M I  OEMBIOS  0x12000527 MSFT 0x00000097) @ 0xfbfff040
ACPI: SRAT (v001 A M I  OEMSRAT  0x12000527 MSFT 0x00000097) @ 0xfbff39b0
ACPI: HPET (v001 A M I  OEMHPET  0x12000527 MSFT 0x00000097) @ 0xfbff3ac0
ACPI: ASF! (v001 AMIASF AMDSTRET 0x00000001 INTL 0x02002026) @ 0xfbff3b00
ACPI: DSDT (v001  0AAAA 0AAAA001 0x00000001 INTL 0x02002026) @ 0x00000000
ACPI: PM-Timer IO Port: 0x5008
ACPI: Local APIC address 0xfee00000
ACPI: LAPIC (acpi_id[0x01] lapic_id[0x00] enabled)
Processor #0 15:5 APIC version 16
ACPI: LAPIC (acpi_id[0x02] lapic_id[0x01] enabled)
Processor #1 15:5 APIC version 16
ACPI: LAPIC (acpi_id[0x03] lapic_id[0x82] disabled)
ACPI: LAPIC (acpi_id[0x04] lapic_id[0x83] disabled)
ACPI: IOAPIC (id[0x02] address[0xfec00000] gsi_base[0])
IOAPIC[0]: apic_id 2, version 17, address 0xfec00000, GSI 0-23
ACPI: IOAPIC (id[0x03] address[0xfebff000] gsi_base[24])
IOAPIC[1]: apic_id 3, version 17, address 0xfebff000, GSI 24-27
ACPI: IOAPIC (id[0x04] address[0xfebfe000] gsi_base[28])
```

```
IOAPIC[2]: apic_id 4, version 17, address 0xfebfe000, GSI 28-31
ACPI: INT_SRC_OVR (bus 0 bus_irq 0 global_irq 2 dfl dfl)
ACPI: INT_SRC_OVR (bus 0 bus_irq 0 global_irq 2 dfl dfl)
ACPI: IRQ0 used by override.
ACPI: IRQ2 used by override.
ACPI: IRQ9 used by override.
Enabling APIC mode:  Flat.  Using 3 I/O APICs
ACPI: HPET id: 0x102282a0 base: 0xfec01000
Using ACPI (MADT) for SMP configuration information
Allocating PCI resources starting at fc400000 (gap: fc000000:03780000)
Built 1 zonelists
Kernel command line: ro root=LABEL=/1 debug panic=5
mapped APIC to ffffd000 (fee00000)
mapped IOAPIC to ffffc000 (fec00000)
mapped IOAPIC to ffffb000 (febff000)
mapped IOAPIC to ffffa000 (febfe000)
Enabling fast FPU save and restore... done.
Enabling unmasked SIMD FPU exception support... done.
Initializing CPU#0
CPU 0 irqstacks, hard=c0565000 soft=c0545000
PID hash table entries: 4096 (order: 12, 65536 bytes)
Console: colour VGA+ 80x25
Dentry cache hash table entries: 131072 (order: 7, 524288 bytes)
Inode-cache hash table entries: 65536 (order: 6, 262144 bytes)
Memory: 8248540k/8388608k available (3118k kernel code, 73068k reserved, 940k data, 288k init, 7405504k highmem)
Checking if this processor honours the WP bit even in supervisor mode... Ok.
Using HPET for base-timer
Using HPET for gettimeofday
Detected 1990.876 MHz processor.
Using hpet for high-res timesource
Calibrating delay using timer specific routine.. 3987.38 BogoMIPS (lpj=7974771)
Mount-cache hash table entries: 512
CPU: After generic identify, caps: 078bfbff e1d3fbff 00000000 00000000 00000000 00000000 00000000
CPU: After vendor identify, caps: 078bfbff e1d3fbff 00000000 00000000 00000000 00000000 00000000
CPU: L1 I Cache: 64K (64 bytes/line), D cache 64K (64 bytes/line)
CPU: L2 Cache: 1024K (64 bytes/line)
CPU: After all inits, caps: 078bfbff e1d3fbff 00000000 00000010 00000000 00000000 00000000
Intel machine check architecture supported.
Intel machine check reporting enabled on CPU#0.
Checking 'hlt' instruction... OK.
CPU0: AMD Opteron(tm) Processor 246 stepping 0a
Booting processor 1/1 eip 2000
CPU 1 irqstacks, hard=c0566000 soft=c0546000
Initializing CPU#1
Calibrating delay using timer specific routine.. 3981.36 BogoMIPS (lpj=7962728)
```

```
CPU: After generic identify, caps: 078bfbff e1d3fbff 00000000 00000000 00000000 00000000 00000000
CPU: After vendor identify, caps: 078bfbff e1d3fbff 00000000 00000000 00000000 00000000 00000000
CPU: L1 I Cache: 64K (64 bytes/line), D cache 64K (64 bytes/line)
CPU: L2 Cache: 1024K (64 bytes/line)
CPU: After all inits, caps: 078bfbff e1d3fbff 00000000 00000010 00000000 00000000 00000000
Intel machine check architecture supported.
Intel machine check reporting enabled on CPU#1.
CPU1: AMD Opteron(tm) Processor 246 stepping 0a
Total of 2 processors activated (7968.74 BogoMIPS).
ENABLING IO-APIC IRQs
..TIMER: vector=0x31 apic1=0 pin1=2 apic2=0 pin2=0
checking TSC synchronization across 2 CPUs: passed.
Brought up 2 CPUs
migration_cost=4000
checking if image is initramfs...it isn't (no cpio magic); looks like an initrd
Freeing initrd memory: 589k freed
NET: Registered protocol family 16
ACPI: bus type pci registered
PCI: PCI BIOS revision 2.10 entry at 0xf0031, last bus=3
PCI: Using configuration type 1
ACPI: Subsystem revision 20060127
ACPI: Interpreter enabled
ACPI: Using IOAPIC for interrupt routing
ACPI: PCI Root Bridge [PCI0] (0000:00)
PCI: Probing PCI hardware (bus 00)
Boot video device is 0000:03:06.0
ACPI: PCI Interrupt Routing Table [\_SB_.PCI0._PRT]
ACPI: PCI Interrupt Routing Table [\_SB_.PCI0.PCI1._PRT]
ACPI: PCI Interrupt Routing Table [\_SB_.PCI0.GOLA._PRT]
ACPI: PCI Interrupt Routing Table [\_SB_.PCI0.GOLB._PRT]
ACPI: PCI Interrupt Link [LNKA] (IRQs 3 4 *5 6 7 9 10 11 12 14 15)
ACPI: PCI Interrupt Link [LNKB] (IRQs 3 4 5 6 7 9 *10 11 12 14 15)
ACPI: PCI Interrupt Link [LNKC] (IRQs 3 4 5 6 7 9 10 *11 12 14 15)
ACPI: PCI Interrupt Link [LNKD] (IRQs 3 4 5 6 7 *9 10 11 12 14 15)
SCSI subsystem initialized
PCI: Using ACPI for IRQ routing
PCI: If a device doesn't work, try "pci=routeirq".  If it helps, post a report
PCI: Bridge: 0000:00:06.0
  IO window: b000-bfff
  MEM window: fca00000-feafffff
  PREFETCH window: disabled.
PCI: Bridge: 0000:00:0a.0
  IO window: disabled.
  MEM window: fc900000-fc9fffff
  PREFETCH window: ff500000-ff5fffff
PCI: Bridge: 0000:00:0b.0
```

```
  IO window: disabled.
  MEM window: fc800000-fc8fffff
  PREFETCH window: ff400000-ff4fffff
highmem bounce pool size: 64 pages
VFS: Disk quotas dquot_6.5.1
Dquot-cache hash table entries: 1024 (order 0, 4096 bytes)
Initializing Cryptographic API
io scheduler noop registered
io scheduler anticipatory registered (default)
io scheduler deadline registered
io scheduler cfq registered
PCI: MSI quirk detected. pci_msi_quirk set.
PCI: MSI quirk detected. pci_msi_quirk set.
pci_hotplug: PCI Hot Plug PCI Core version: 0.5
Real Time Clock Driver v1.12ac
serio: i8042 AUX port at 0x60,0x64 irq 12
serio: i8042 KBD port at 0x60,0x64 irq 1
Serial: 8250/16550 driver $Revision: 1.90 $ 4 ports, IRQ sharing disabled
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
serial8250: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
RAMDISK driver initialized: 16 RAM disks of 16384K size 1024 blocksize
Compaq SMART2 Driver (v 2.6.0)
HP CISS Driver (v 2.6.10)
Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 33MHz system bus speed for PIO modes; override with idebus=xx
AMD8111: IDE controller at PCI slot 0000:00:07.1
AMD8111: chipset revision 3
AMD8111: not 100% native mode: will probe irqs later
AMD8111: 0000:00:07.1 (rev 03) UDMA133 controller
    ide0: BM-DMA at 0xffa0-0xffa7, BIOS settings: hda:pio, hdb:pio
    ide1: BM-DMA at 0xffa8-0xffaf, BIOS settings: hdc:pio, hdd:pio
Probing IDE interface ide0...
Probing IDE interface ide1...
Probing IDE interface ide0...
Probing IDE interface ide1...
Adaptec aacraid driver (1.1-4 May  4 2006 17:43:05)
Emulex LightPulse Fibre Channel SCSI driver 8.1.1
Copyright(c) 2004-2005 Emulex.  All rights reserved.
megaraid cmm: 2.20.2.6 (Release Date: Mon Mar 7 00:01:03 EST 2005)
megaraid: 2.20.4.7 (Release Date: Mon Nov 14 12:27:22 EST 2005)
megaraid: probe new device 0x1000:0x1960:0x1000:0x4523: bus 1:slot 4:func 0
ACPI: PCI Interrupt 0000:01:04.0[A] -> GSI 29 (level, low) -> IRQ 16
megaraid: fw version:[713N] bios version:[G119]
scsi0 : LSI Logic MegaRAID driver
scsi[0]: scanning scsi channel 0 [Phy 0] for non-raid devices
scsi[0]: scanning scsi channel 1 [virtual] for logical drives
  Vendor: MegaRAID  Model: LD 0 RAID1  476G  Rev: 713N
  Type:   Direct-Access                      ANSI SCSI revision: 02
```

```
GDT-HA: Storage RAID Controller Driver. Version: 3.04
GDT-HA: Found 0 PCI Storage RAID Controllers
3ware Storage Controller device driver for Linux v1.26.02.001.
3ware 9000 Storage Controller device driver for Linux v2.26.02.005.
libata version 1.20 loaded.
SCSI device sda: 976773120 512-byte hdwr sectors (500108 MB)
sda: Write Protect is off
sda: Mode Sense: 00 00 00 00
sda: asking for cache data failed
sda: assuming drive cache: write through
SCSI device sda: 976773120 512-byte hdwr sectors (500108 MB)
sda: Write Protect is off
sda: Mode Sense: 00 00 00 00
sda: asking for cache data failed
sda: assuming drive cache: write through
 sda: sda1 sda2 sda3 sda4 < sda5 sda6 >
sd 0:1:0:0: Attached scsi disk sda
mice: PS/2 mouse device common for all mice
md: linear personality registered for level -1
md: raid0 personality registered for level 0
md: raid1 personality registered for level 1
md: raid10 personality registered for level 10
md: raid5 personality registered for level 5
md: raid4 personality registered for level 4
raid5: automatically using best checksumming function: pIII_sse
   pIII_sse  :  6405.000 MB/sec
raid5: using function: pIII_sse (6405.000 MB/sec)
md: multipath personality registered for level -4
md: md driver 0.90.3 MAX_MD_DEVS=256, MD_SB_DISKS=27
md: bitmap version 4.39
device-mapper: 4.5.0-ioctl (2005-10-04) initialised: dm-devel@redhat.com
device-mapper: dm-multipath version 1.0.4 loaded
device-mapper: dm-round-robin version 1.0.0 loaded
device-mapper: dm-emc version 0.0.3 loaded
NET: Registered protocol family 2
IP route cache hash table entries: 524288 (order: 9, 2097152 bytes)
TCP established hash table entries: 524288 (order: 10, 4194304 bytes)
TCP bind hash table entries: 65536 (order: 7, 524288 bytes)
TCP: Hash tables configured (established 524288 bind 65536)
TCP reno registered
TCP bic registered
NET: Registered protocol family 1
Starting balanced_irq
Using IPI Shortcut mode
md: Autodetecting RAID arrays.
md: autorun ...
md: ... autorun DONE.
RAMDISK: Compressed image found at block 0
```

```
VFS: Mounted root (ext2 filesystem).
kjournald starting.  Commit interval 5 seconds
EXT3-fs: mounted filesystem with ordered data mode.
Freeing unused kernel memory: 288k freed
floppy0: no floppy controllers found
tg3.c:v3.49 (Feb 2, 2006)
ACPI: PCI Interrupt 0000:02:09.0[A] -> GSI 24 (level, low) -> IRQ 17
eth0: Tigon3 [partno(BCM95704A7) rev 2003 PHY(5704)] (PCIX:100MHz:64-bit)
10/100/1000BaseT Ethernet 00:e0:81:2f:90:96
eth0: RXcsums[1] LinkChgREG[0] MIirq[0] ASF[0] Split[0] WireSpeed[1] TSOcap[1]
eth0: dma_rwctrl[769f4000] dma_mask[64-bit]
ACPI: PCI Interrupt 0000:02:09.1[B] -> GSI 25 (level, low) -> IRQ 18
eth1: Tigon3 [partno(BCM95704A7) rev 2003 PHY(5704)] (PCIX:100MHz:64-bit)
10/100/1000BaseT Ethernet 00:e0:81:2f:90:97
eth1: RXcsums[1] LinkChgREG[0] MIirq[0] ASF[0] Split[0] WireSpeed[1] TSOcap[1]
eth1: dma_rwctrl[769f4000] dma_mask[64-bit]
shpchp: HPC vendor_id 1022 device_id 7460 ss_vid 0 ss_did 0
shpchp: shpc_init: cannot reserve MMIO region
shpchp: HPC vendor_id 1022 device_id 7450 ss_vid 0 ss_did 0
shpchp: shpc_init: cannot reserve MMIO region
shpchp: HPC vendor_id 1022 device_id 7450 ss_vid 0 ss_did 0
shpchp: shpc_init: cannot reserve MMIO region
shpchp: Standard Hot Plug PCI Controller Driver version: 0.4
usbcore: registered new driver usbfs
usbcore: registered new driver hub
ohci_hcd: 2005 April 22 USB 1.1 'Open' Host Controller (OHCI) Driver (PCI)
ACPI: PCI Interrupt 0000:03:00.0[D] -> GSI 19 (level, low) -> IRQ 19
ohci_hcd 0000:03:00.0: OHCI Host Controller
ohci_hcd 0000:03:00.0: new USB bus registered, assigned bus number 1
ohci_hcd 0000:03:00.0: irq 19, io mem 0xfeafc000
usb usb1: configuration #1 chosen from 1 choice
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 3 ports detected
ACPI: PCI Interrupt 0000:03:00.1[D] -> GSI 19 (level, low) -> IRQ 19
ohci_hcd 0000:03:00.1: OHCI Host Controller
ohci_hcd 0000:03:00.1: new USB bus registered, assigned bus number 2
ohci_hcd 0000:03:00.1: irq 19, io mem 0xfeafd000
usb usb2: configuration #1 chosen from 1 choice
hub 2-0:1.0: USB hub found
hub 2-0:1.0: 3 ports detected
md: Autodetecting RAID arrays.
md: autorun ...
md: ... autorun DONE.
ACPI: Power Button (FF) [PWRF]
ACPI: Power Button (CM) [PWRB]
ACPI: Processor [CPU1] (supports 8 throttling states)
EXT3 FS on sda2, internal journal
program dmraid is using a deprecated SCSI ioctl, please convert it to SG_IO
```

```
kjournald starting.  Commit interval 5 seconds
EXT3 FS on sda1, internal journal
EXT3-fs: mounted filesystem with ordered data mode.
kjournald starting.  Commit interval 5 seconds
EXT3 FS on sda3, internal journal
EXT3-fs: mounted filesystem with ordered data mode.
kjournald starting.  Commit interval 5 seconds
EXT3 FS on sda6, internal journal
EXT3-fs: mounted filesystem with ordered data mode.
Adding 4192924k swap on /dev/sda5.  Priority:-1 extents:1 across:4192924k
NET: Registered protocol family 17
tg3: eth0: Link is up at 1000 Mbps, full duplex.
tg3: eth0: Flow control is off for TX and off for RX.
lp: driver loaded but no devices found
sd_init_command: disk=sda, block=8596831, count=8
sda : block=8596831
sda : reading 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30980, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30980                 sd 0:1:0:0:
        command: Read (10): 28 00 00 83 2d 5f 00 00 08 00
buffer = 0xf7cfb240, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30980, rtn: 1
sd 0:1:0:0: done 0xf7f30980 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 83 2d 5f 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=25372896, count=2
sda : block=25372896
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                 sd 0:1:0:0:
        command: Read (10): 28 00 01 83 28 e0 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=2992845, count=8
sda : block=2992845
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30500, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30500                 sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d aa cd 00 00 08 00
buffer = 0xf6bfeb00, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 01 83 28 e0 00 00 02 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=446976176, count=2
sda : block=446976176
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 1a a4 50 b0 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30500, rtn: 1
sd 0:1:0:0: done 0xf7f30500 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d aa cd 00 00 08 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a4 50 b0 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=237405, count=16
sda : block=237405
sda : writing 16/16 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 9f 5d 00 00 10 00
buffer = 0xf6bfebc0, bufflen = 8192, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 9f 5d 00 00 10 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
16 sectors total, 8192 bytes done.
use_sg is 2
sd_init_command: disk=sda, block=446984364, count=2
sda : block=446984364
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 1a a4 70 ac 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a4 70 ac 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447205552, count=2
sda : block=447205552
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 1a a7 d0 b0 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a7 d0 b0 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447217836, count=2
sda : block=447217836
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 1a a8 00 ac 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a8 00 ac 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447205554, count=2
sda : block=447205554
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380             sd 0:1:0:0:
        command: Read (10): 28 00 1a a7 d0 b2 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a7 d0 b2 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447218934, count=2
sda : block=447218934
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380             sd 0:1:0:0:
        command: Read (10): 28 00 1a a8 04 f6 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a a8 04 f6 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447811834, count=2
sda : block=447811834
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380             sd 0:1:0:0:
        command: Read (10): 28 00 1a b1 10 fa 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a b1 10 fa 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=447825518, count=2
sda : block=447825518
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 1a b1 46 6e 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1a b1 46 6e 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=871764144, count=2
sda : block=871764144
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 10 b0 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 10 b0 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=871769260, count=26
sda : block=871769260
sda : reading 26/26 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
buffer = 0xf6bfebc0, bufflen = 13312, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
26 sectors total, 13312 bytes done.
use_sg is 3
attempt to access beyond end of device
sda6: rw=0, want=1044134458, limit=951401367
```

```
Buffer I/O error on device sda6, logical block 522067228
attempt to access beyond end of device
sda6: rw=0, want=1178878530, limit=951401367
Buffer I/O error on device sda6, logical block 589439264
attempt to access beyond end of device
sda6: rw=0, want=1313622602, limit=951401367
Buffer I/O error on device sda6, logical block 656811300
attempt to access beyond end of device
sda6: rw=0, want=1448366674, limit=951401367
Buffer I/O error on device sda6, logical block 724183336
attempt to access beyond end of device
sda6: rw=0, want=1583110746, limit=951401367
Buffer I/O error on device sda6, logical block 791555372
attempt to access beyond end of device
sda6: rw=0, want=1717854818, limit=951401367
Buffer I/O error on device sda6, logical block 858927408
attempt to access beyond end of device
sda6: rw=0, want=1852598890, limit=951401367
Buffer I/O error on device sda6, logical block 926299444
attempt to access beyond end of device
sda6: rw=0, want=1987342962, limit=951401367
Buffer I/O error on device sda6, logical block 993671480
attempt to access beyond end of device
sda6: rw=0, want=2122087034, limit=951401367
Buffer I/O error on device sda6, logical block 1061043516
attempt to access beyond end of device
sda6: rw=0, want=2256831106, limit=951401367
Buffer I/O error on device sda6, logical block 1128415552
attempt to access beyond end of device
sda6: rw=0, want=2391575178, limit=951401367
attempt to access beyond end of device
sda6: rw=0, want=2526319250, limit=951401367
attempt to access beyond end of device
sda6: rw=0, want=2661063322, limit=951401367
sd_init_command: disk=sda, block=934757082, count=2
sda : block=934757082
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                 sd 0:1:0:0:
        command: Read (10): 28 00 37 b7 42 da 00 00 02 00
buffer = 0xf6bfebc0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=126292650, count=2
sda : block=126292650
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30500, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30500                 sd 0:1:0:0:
        command: Read (10): 28 00 07 87 12 aa 00 00 02 00
```

```
buffer = 0xf6bfeb00, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=261036722, count=2
sda : block=261036722
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30680, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30680            sd 0:1:0:0:
        command: Read (10): 28 00 0f 8f 1a b2 00 00 02 00
buffer = 0xf6bfea40, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=395780794, count=2
sda : block=395780794
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30800, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30800            sd 0:1:0:0:
        command: Read (10): 28 00 17 97 22 ba 00 00 02 00
buffer = 0xf6bfe980, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=530524866, count=2
sda : block=530524866
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30e00, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30e00            sd 0:1:0:0:
        command: Read (10): 28 00 1f 9f 2a c2 00 00 02 00
buffer = 0xf6bfe8c0, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=665268938, count=2
sda : block=665268938
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30c80, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30c80            sd 0:1:0:0:
        command: Read (10): 28 00 27 a7 32 ca 00 00 02 00
buffer = 0xf6bfe800, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=800013010, count=2
sda : block=800013010
sda : reading 2/2 512 byte blocks.
scsi_add_timer: scmd: f7f30b00, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30b00            sd 0:1:0:0:
        command: Read (10): 28 00 2f af 3a d2 00 00 02 00
buffer = 0xf6bfe740, bufflen = 1024, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
attempt to access beyond end of device
sda6: rw=0, want=2795807394, limit=951401367
attempt to access beyond end of device
sda6: rw=0, want=2930551466, limit=951401367
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 37 b7 42 da 00 00 02 00
scsi host busy 7 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30500, rtn: 1
sd 0:1:0:0: done 0xf7f30500 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 07 87 12 aa 00 00 02 00
scsi host busy 6 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30680, rtn: 1
sd 0:1:0:0: done 0xf7f30680 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 0f 8f 1a b2 00 00 02 00
scsi host busy 5 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30800, rtn: 1
sd 0:1:0:0: done 0xf7f30800 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 17 97 22 ba 00 00 02 00
scsi host busy 4 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30e00, rtn: 1
sd 0:1:0:0: done 0xf7f30e00 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 1f 9f 2a c2 00 00 02 00
scsi host busy 3 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30c80, rtn: 1
sd 0:1:0:0: done 0xf7f30c80 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 27 a7 32 ca 00 00 02 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30b00, rtn: 1
sd 0:1:0:0: done 0xf7f30b00 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 2f af 3a d2 00 00 02 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
2 sectors total, 1024 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=237421, count=8
sda : block=237421
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30b00, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30b00                   sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 9f 6d 00 00 08 00
buffer = 0xf6bfe740, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
attempt to access beyond end of device
sda6: rw=0, want=1044134458, limit=951401367
scsi_delete_timer: scmd: f7f30b00, rtn: 1
sd 0:1:0:0: done 0xf7f30b00 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 9f 6d 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=8338293, count=8
sda : block=8338293
sda : reading 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30c80, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30c80                   sd 0:1:0:0:
        command: Read (10): 28 00 00 7f 3b 75 00 00 08 00
buffer = 0xf6bfebc0, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30c80, rtn: 1
sd 0:1:0:0: done 0xf7f30c80 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 7f 3b 75 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=8382885, count=64
sda : block=8382885
sda : reading 64/64 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                   sd 0:1:0:0:
        command: Read (10): 28 00 00 7f e9 a5 00 00 40 00
buffer = 0xf6bfeb00, bufflen = 32768, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
```

```
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 7f e9 a5 00 00 40 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
64 sectors total, 32768 bytes done.
use_sg is 8
sd_init_command: disk=sda, block=8382949, count=40
sda : block=8382949
sda : reading 40/40 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                  sd 0:1:0:0:
        command: Read (10): 28 00 00 7f e9 e5 00 00 28 00
buffer = 0xf6bfeb00, bufflen = 20480, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=2992901, count=8
sda : block=2992901
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30c80, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30c80                  sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d ab 05 00 00 08 00
buffer = 0xf6bfebc0, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 7f e9 e5 00 00 28 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
40 sectors total, 20480 bytes done.
use_sg is 4
sd_init_command: disk=sda, block=8382989, count=16
sda : block=8382989
sda : reading 16/16 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                  sd 0:1:0:0:
        command: Read (10): 28 00 00 7f ea 0d 00 00 10 00
buffer = 0xf6bfeb00, bufflen = 8192, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 7f ea 0d 00 00 10 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
16 sectors total, 8192 bytes done.
use_sg is 2
```

```
scsi_delete_timer: scmd: f7f30c80, rtn: 1
sd 0:1:0:0: done 0xf7f30c80 SUCCESS        0 sd 0:1:0:0: 
        command: Write (10): 2a 00 00 2d ab 05 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=247085, count=16
sda : block=247085
sda : writing 16/16 512 byte blocks.
scsi_add_timer: scmd: f7f30c80, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30c80                  sd 0:1:0:0: 
        command: Write (10): 2a 00 00 03 c5 2d 00 00 10 00
buffer = 0xf6bfebc0, bufflen = 8192, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30c80, rtn: 1
sd 0:1:0:0: done 0xf7f30c80 SUCCESS        0 sd 0:1:0:0: 
        command: Write (10): 2a 00 00 03 c5 2d 00 00 10 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
16 sectors total, 8192 bytes done.
use_sg is 2
sd 0:1:0:0: Attached scsi generic sg0 type 0
sg_open: dev=0, flags=0x8002
scsi_block_when_processing_errors: rtn: 1
sg_add_sfp: sfp=0xf680e000
sg_build_reserve: req_size=32768
sg_build_indirect: buff_size=32768, blk_size=32768
sg_build_build: k=0, a=0xc16d0900, len=32768
sg_build_indirect: k_use_sg=1, rem_sz=0
sg_add_sfp:   bufflen=32768, k_use_sg=1
sg_ioctl: sg0, cmd=0x2282
sd_init_command: disk=sda, block=2992917, count=8
sda : block=2992917
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30800, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30800                  sd 0:1:0:0: 
        command: Write (10): 2a 00 00 2d ab 15 00 00 08 00
buffer = 0xf6bfe080, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sg_ioctl: sg0, cmd=0x2275
sg_remove_scat: k_use_sg=1
sg_remove_scat: k=0, a=0xc16d0900, len=32768
sg_build_reserve: req_size=65536
sg_build_indirect: buff_size=65536, blk_size=65536
sg_build_build: k=0, a=0xc16d0900, len=32768
```

```
sg_build_build: k=1, a=0xc16d0a00, len=32768
sg_build_indirect: k_use_sg=2, rem_sz=0
sg_ioctl: sg0, cmd=0x227b
sg_ioctl: sg0, cmd=0x2276
sg_write: sg0, count=64
scsi_block_when_processing_errors: rtn: 1
sg_common_write:  scsi opcode=0x28, cmd_size=10
sg_start_req: dxfer_len=13312
sg_link_reserve: size=13312
scsi_add_timer: scmd: f7f30b00, time: 15000, (c02b1420)
sd 0:1:0:0: send 0xf7f30b00             sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
buffer = 0xf6bfe680, bufflen = 13312, done = 0xc02b4b90, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sg_read: sg0, count=64
scsi_delete_timer: scmd: f7f30800, rtn: 1
sd 0:1:0:0: done 0xf7f30800 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d ab 15 00 00 08 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=249717, count=56
sda : block=249717
sda : writing 56/56 512 byte blocks.
scsi_add_timer: scmd: f7f30800, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30800             sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 cf 75 00 00 38 00
buffer = 0xf6bfe080, bufflen = 28672, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30b00, rtn: 1
sd 0:1:0:0: done 0xf7f30b00 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 33 f6 24 ac 00 00 1a 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
26 sectors total, 13312 bytes done.
use_sg is 1
sg_cmd_done: sg0, pack_id=871769260, res=0x0
scsi_delete_timer: scmd: f7f30800, rtn: 1
sd 0:1:0:0: done 0xf7f30800 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 cf 75 00 00 38 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
56 sectors total, 28672 bytes done.
use_sg is 7
sd_init_command: disk=sda, block=249773, count=8
```

```
sda : block=249773
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30800, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30800                sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 cf ad 00 00 08 00
buffer = 0xf6bfe080, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sg_read_xfer: num_xfer=13312, iovec_count=0, k_use_sg=1
scsi_delete_timer: scmd: f7f30800, rtn: 1
sd 0:1:0:0: done 0xf7f30800 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 03 cf ad 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
sg_finish_rem_req: res_used=1
sg_unlink_reserve: req->k_use_sg=1
sd_init_command: disk=sda, block=2992917, count=8
sda : block=2992917
sda : writing 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30800, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30800                sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d ab 15 00 00 08 00
buffer = 0xf6bfe080, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
sd_init_command: disk=sda, block=8807823, count=8
sda : block=8807823
sda : reading 8/8 512 byte blocks.
scsi_add_timer: scmd: f7f30b00, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30b00                sd 0:1:0:0:
        command: Read (10): 28 00 00 86 65 8f 00 00 08 00
buffer = 0xf6bfe680, bufflen = 4096, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30800, rtn: 1
sd 0:1:0:0: done 0xf7f30800 SUCCESS        0 sd 0:1:0:0:
        command: Write (10): 2a 00 00 2d ab 15 00 00 08 00
scsi host busy 2 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
8 sectors total, 4096 bytes done.
use_sg is 1
scsi_delete_timer: scmd: f7f30b00, rtn: 1
sd 0:1:0:0: done 0xf7f30b00 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 86 65 8f 00 00 08 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
```

```
8 sectors total, 4096 bytes done.
use_sg is 1
sd_init_command: disk=sda, block=4232917, count=64
sda : block=4232917
sda : reading 64/64 512 byte blocks.
scsi_add_timer: scmd: f7f30380, time: 7500, (c02b1420)
sd 0:1:0:0: send 0xf7f30380                sd 0:1:0:0:
        command: Read (10): 28 00 00 40 96 d5 00 00 40 00
buffer = 0xf6bfea40, bufflen = 32768, done = 0xc0366b40, queuecommand 0xc0344010
leaving scsi_dispatch_cmnd()
scsi_delete_timer: scmd: f7f30380, rtn: 1
sd 0:1:0:0: done 0xf7f30380 SUCCESS        0 sd 0:1:0:0:
        command: Read (10): 28 00 00 40 96 d5 00 00 40 00
scsi host busy 1 failed 0
sd 0:1:0:0: Notifying upper driver of completion (result 0)
sd_rw_intr: sda: res=0x0
64 sectors total, 32768 bytes done.
use_sg is 5
sg_release: sg0
sg_fasync: sg0, mode=0
__sg_remove_sfp:    bufflen=65536, k_use_sg=2
sg_remove_scat: k_use_sg=2
sg_remove_scat: k=0, a=0xc16d0900, len=32768
sg_remove_scat: k=1, a=0xc16d0a00, len=32768
__sg_remove_sfp:    sfp=0xf680e000
```

File Attachments

1) ttt, downloaded 281 times
2) ttt.sgp, downloaded 260 times

Subject: Re: megaraid_mbox: garbage in file

Posted by James Bottomley on Fri, 05 May 2006 15:59:22 GMT
View Forum Message <> Reply to Message

On Fri, 2006-05-05 at 09:37 +0400, Vasily Averin wrote:

> The issue is that the correctly finished scsi read command return me garbage
> (repeated 0 ...127 -- see hexdump in my first letter) instead correct file content.
> "attempt to access beyond end of device" messages occurs due the same garbage
> readed from the Indirect block. I found this garbage present in data buffers
> beginning at megaraid driver functions.
>
> I would note that if I read the same file by using dd with bs=1024 or bs=512 --
> I get correct file content.
>
> When I use kernel with 4Gb memory limit -- the same cat command return me
> correct file content too, without any garbage.
>

> Question is what it is the strange garbage? Have you seen it earlier?
> Is it possible that it is some driver-related issue or it is broken hardware?
> And why I can workaround this issue by using only 4Gb memory?

This is really odd ... if the controller can't reach *any* memory above
32 bits, then, on an 8GB machine you'd expect corruption all over the
place since most user pages come from the top of highmem.

The first thing to try, since you have an opteron system, is to get rid
of highmem entirely and use a 64 bit kernel (just to make sure we're not
running into some annoying dma_addr_t conversion problem). Then, I
suppose if that doesn't work, try printing out the actual contents of
the sg list to see what the physical memory location of the page
containing the corrupt block is.

This could also be a firmware problem, I suppose, but I haven't seen any
similar reports.

James

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 05 May 2006 18:14:25 GMT

View Forum Message <> Reply to Message

James Bottomley wrote:
> On Fri, 2006-05-05 at 09:37 +0400, Vasily Averin wrote:
>>The issue is that the correctly finished scsi read command return me garbage
>>(repeated 0 ...127 -- see hexdump in my first letter) instead correct file content.
>>"attempt to access beyond end of device" messages occurs due the same garbage
>>readed from the Indirect block. I found this garbage present in data buffers
>>beginning at megaraid driver functions.
>>
>>I would note that if I read the same file by using dd with bs=1024 or bs=512 --
>>I get correct file content.
>>
>>When I use kernel with 4Gb memory limit -- the same cat command return me
>>correct file content too, without any garbage.
>>
>>Question is what it is the strange garbage? Have you seen it earlier?
>>Is it possible that it is some driver-related issue or it is broken hardware?
>>And why I can workaround this issue by using only 4Gb memory?
>
> This is really odd ... if the controller can't reach *any* memory above
> 32 bits, then, on an 8GB machine you'd expect corruption all over the
> place since most user pages come from the top of highmem.
>
> The first thing to try, since you have an opteron system, is to get rid

> of highmem entirely and use a 64 bit kernel (just to make sure we're not
> running into some annoying dma_addr_t conversion problem).

Unfortunately it is customers node, and I'm not able to re-install 64-bit distribution to load 64-bit kernel. Of course I'll ask customer about this, but it will be done later.

> Then, I
> suppose if that doesn't work, try printing out the actual contents of
> the sg list to see what the physical memory location of the page
> containing the corrupt block is.

I've already done such experiment:
On 2.6.8-based virtuozzo kernel I've added following code to megaraid_mbox_display_scb function:

```
  virt = page_address(sg[i].page) + sg[i].offset;
  printk("mbox sg%d: page %p off %d addr %llx len %d "
         "virt %p first %08x page->flags %08x\n",
   i, sg[i].page, sg[i].offset, sg[i].dma_address, sg[i].length,
   virt, virt == NULL ? 0: *(int *)virt, sg[i].page->flags);
```

and get the following results

```
May  4 02:51:38 vpsn002 kernel:
 megaraid mailbox: status:0x0 cmd:0xa7 id:0x25 sec:0x1a
   lba:0x33f624ac addr:0xffffffff ld:128 sg:4
 scsi cmnd: 0x28 0x00 0x33 0xf6 0x24 0xac 0x00 0x00 0x1a 0x00
 mbox request_buffer eafde340 use_sg 4
 mbox sg0: page 077a0474 off 0 addr 1fd575000 len 4096 virt ff15a000
   first 03020100 page->flags 40020101
 mbox sg1: page 077b5738 off 0 addr 1fdede000 len 4096 virt ff141000
   first 03020100 page->flags 40020101
 mbox sg2: page 077ad500 off 0 addr 1fdb40000 len 4096 virt ff056000
   first 03020100 page->flags 40020101
 mbox sg3: page 030d46e8 off 1024 addr 5e6a400 len 1024 virt 07e6a400
   first 03020100 page->flags 20001004
```

"first 03020100" shows that data in the all sg buffers is already corrupted.
Also I would note that page for last 1Kb buffer is not Highmem.

If you want I can reproduce this experiment on 2.6.16 kernel too.

> This could also be a firmware problem, I suppose, but I haven't seen any
> similar reports.

Thank you,
 Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: RE: megaraid_mbox: garbage in file
Posted by Seokmann.Ju on Fri, 05 May 2006 19:59:24 GMT
View Forum Message <> Reply to Message

Can you do one quick change in the driver?
Search for 'pci_set_dma_mask()' API calls in the driver and mask out one of them with DMA_64BIT_MASK as follow.

```
---
 // if (pci_set_dma_mask(adapter->pdev, DMA_64BIT_MASK) != 0) {

 //  conlog(CL_ANN, (KERN_WARNING
 //   "megaraid: could not set DMA mask for 64-bit.\n"));

 //  goto out_free_sysfs_res;
 // }
---
```

I found that the driver is NOT checking 64-bit DMA capability of the controllers accordingly and this could be a reason.
I'm waiting for feedback from F/W team for MegaRAID 150-4 controller if it supports 64-bit DMA.

I'll update here as I get.

Thank you,

> -----Original Message-----
> From: Vasily Averin [mailto:vvs@sw.ru]
> Sent: Friday, May 05, 2006 2:17 PM
> To: James Bottomley
> Cc: linux-scsi@vger.kernel.org; Kolli, Neela; Mukker, Atul;
> Ju, Seokmann; Bagalkote, Sreenivas; devel@openvz.org; Linux
> Kernel Mailing List
> Subject: Re: megaraid_mbox: garbage in file
>
> James Bottomley wrote:
> > On Fri, 2006-05-05 at 09:37 +0400, Vasily Averin wrote:
> >>The issue is that the correctly finished scsi read command
> return me garbage
> >>(repeated 0 ...127 -- see hexdump in my first letter)
> instead correct file content.
> >>"attempt to access beyond end of device" messages occurs
> due the same garbage
> >>readed from the Indirect block. I found this garbage
> present in data buffers
> >>beginning at megaraid driver functions.
> >>
> >>I would note that if I read the same file by using dd with
> bs=1024 or bs=512 --
> >>I get correct file content.

> >>
> >>When I use kernel with 4Gb memory limit -- the same cat
> command return me
> >>correct file content too, without any garbage.
> >>
> >>Question is what it is the strange garbage? Have you seen
> it earlier?
> >>Is it possible that it is some driver-related issue or it
> is broken hardware?
> >>And why I can workaround this issue by using only 4Gb memory?
> >
> > This is really odd ... if the controller can't reach *any*
> memory above
> > 32 bits, then, on an 8GB machine you'd expect corruption
> all over the
> > place since most user pages come from the top of highmem.
> >
> > The first thing to try, since you have an opteron system,
> is to get rid
> > of highmem entirely and use a 64 bit kernel (just to make
> sure we're not
> > running into some annoying dma_addr_t conversion problem).
>
> Unfortunately it is customers node, and I'm not able to
> re-install 64-bit
> distribution to load 64-bit kernel. Of course I'll ask
> customer about this, but
> it will be done later.
>
> > Then, I
> > suppose if that doesn't work, try printing out the actual
> contents of
> > the sg list to see what the physical memory location of the page
> > containing the corrupt block is.
>
> I've already done such experiment:
> On 2.6.8-based virtuozzo kernel I've added following code to
> megaraid_mbox_display_scb function:

```
>   virt = page_address(sg[i].page) + sg[i].offset;
>   printk("mbox sg%d: page %p off %d addr %llx len %d "
>          "virt %p first %08x page->flags %08x\n",
>    i, sg[i].page, sg[i].offset, sg[i].dma_address, sg[i].length,
>    virt, virt == NULL ? 0: *(int *)virt, sg[i].page->flags);
>
```

> and get the following results

```
> May  4 02:51:38 vpsn002 kernel:
>  megaraid mailbox: status:0x0 cmd:0xa7 id:0x25 sec:0x1a
>    lba:0x33f624ac addr:0xffffffff ld:128 sg:4
>  scsi cmnd: 0x28 0x00 0x33 0xf6 0x24 0xac 0x00 0x00 0x1a 0x00
>  mbox request_buffer eafde340 use_sg 4
>  mbox sg0: page 077a0474 off 0 addr 1fd575000 len 4096 virt ff15a000
>    first 03020100 page->flags 40020101
>  mbox sg1: page 077b5738 off 0 addr 1fdede000 len 4096 virt ff141000
>    first 03020100 page->flags 40020101
>  mbox sg2: page 077ad500 off 0 addr 1fdb40000 len 4096 virt ff056000
>    first 03020100 page->flags 40020101
>  mbox sg3: page 030d46e8 off 1024 addr 5e6a400 len 1024 virt 07e6a400
>    first 03020100 page->flags 20001004
>
```

> "first 03020100" shows that data in the all sg buffers is
> already corrupted.
> Also I would note that page for last 1Kb buffer is not Highmem.
>
> If you want I can reproduce this experiment on 2.6.16 kernel too.
>
> > This could also be a firmware problem, I suppose, but I
> haven't seen any
> > similar reports.
>
> Thank you,
> Vasily Averin
>
> SWsoft Virtuozzo/OpenVZ Linux kernel team
>

Subject: Re: megaraid_mbox: garbage in file
Posted by James Bottomley on Fri, 05 May 2006 20:05:09 GMT
View Forum Message <> Reply to Message

On Fri, 2006-05-05 at 22:17 +0400, Vasily Averin wrote:

```
>  megaraid mailbox: status:0x0 cmd:0xa7 id:0x25 sec:0x1a
>                 lba:0x33f624ac addr:0xffffffff ld:128 sg:4
>  scsi cmnd: 0x28 0x00 0x33 0xf6 0x24 0xac 0x00 0x00 0x1a 0x00
>  mbox request_buffer eafde340 use_sg 4
>  mbox sg0: page 077a0474 off 0 addr 1fd575000 len 4096 virt ff15a000
>                 first 03020100 page->flags 40020101
>  mbox sg1: page 077b5738 off 0 addr 1fdede000 len 4096 virt ff141000
>                 first 03020100 page->flags 40020101
>  mbox sg2: page 077ad500 off 0 addr 1fdb40000 len 4096 virt ff056000
>                 first 03020100 page->flags 40020101
>  mbox sg3: page 030d46e8 off 1024 addr 5e6a400 len 1024 virt 07e6a400
>                 first 03020100 page->flags 20001004
```

The odd thing about this is that the highmem addresses shouldn't have a
virtual mapping (since nothing should have called kmap on them).

But the other tickles a suspicion about the card. I know various LSI
chips that don't have a true 64 bit mode, but instead have programmable
windowed mappings in their descriptors (i.e. all SG list elements have
to be in the same xGB region of physical memory), and since the last
descriptor is more than 4GB away from the other three, whether this
might be the problem here. Unfortunately, only LSI can tell us this ...

James

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 05 May 2006 23:32:48 GMT

View Forum Message <> Reply to Message

Ju, Seokmann wrote:

```
> Can you do one quick change in the driver?
> Search for 'pci_set_dma_mask()' API calls in the driver and mask out one of them with DMA_64BIT_MASK as follow.
> ---
>  // if (pci_set_dma_mask(adapter->pdev, DMA_64BIT_MASK) != 0) {
>
>  //  conlog(CL_ANN, (KERN_WARNING
>  //   "megaraid: could not set DMA mask for 64-bit.\n"));
>
>  //  goto out_free_sysfs_res;
>  // }
> ---
>
> I found that the driver is NOT checking 64-bit DMA capability of the controllers accordingly and this could be a reason.
```

This change help me:

```
megaraid mailbox: status:0x0 cmd:0xa7 id:0x1f sec:0x1a lba:0x33f624ac
addr:0xffffffff ld:128 sg:4
scsi cmnd: 0x28 0x00 0x33 0xf6 0x24 0xac 0x00 0x00 0x1a 0x00
mbox request_buffer ebeb9380 use_sg 4
mbox sg0: page 050c5d88 off 0 addr e90d2000 len 4096 virt eb0d2000
  first 732e646c page->flags 20000000
mbox sg1: page 050c5710 off 0 addr e90a4000 len 4096 virt eb0a4000
  first 00000003 page->flags 20000000
mbox sg2: page 050c4438 off 0 addr e901e000 len 4096 virt eb01e000
  first 00000000 page->flags 20000000
mbox sg3: page 030d64dc off 1024 addr 5f3f400 len 1024 virt 07f3f400
  first 19398a0e page->flags 20001004
```

Errors go away, file content is correct.

> I'm waiting for feedback from F/W team for MegaRAID 150-4 controller if it supports 64-bit DMA.
>
> I'll update here as I get.

How do you this, can it be the cause of
http://bugzilla.kernel.org/show_bug.cgi?id=6052

If so, you have a good chance to resolve this bug too :)

Thank you,
 Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 05 May 2006 23:40:25 GMT

View Forum Message <> Reply to Message

James Bottomley wrote:

```
> On Fri, 2006-05-05 at 22:17 +0400, Vasily Averin wrote:
>> megaraid mailbox: status:0x0 cmd:0xa7 id:0x25 sec:0x1a
>>               lba:0x33f624ac addr:0xffffffff ld:128 sg:4
>> scsi cmnd: 0x28 0x00 0x33 0xf6 0x24 0xac 0x00 0x00 0x1a 0x00
>> mbox request_buffer eafde340 use_sg 4
>> mbox sg0: page 077a0474 off 0 addr 1fd575000 len 4096 virt ff15a000
>>               first 03020100 page->flags 40020101
>> mbox sg1: page 077b5738 off 0 addr 1fdede000 len 4096 virt ff141000
>>               first 03020100 page->flags 40020101
>> mbox sg2: page 077ad500 off 0 addr 1fdb40000 len 4096 virt ff056000
>>               first 03020100 page->flags 40020101
>> mbox sg3: page 030d46e8 off 1024 addr 5e6a400 len 1024 virt 07e6a400
>>               first 03020100 page->flags 20001004
>
> The odd thing about this is that the highmem addresses shouldn't have a
> virtual mapping (since nothing should have called kmap on them).
```

You are right, in the other my experiments highmem pages usually have virt=0 and
I cannot find who is kmapped these pages.

on 2.6.16 kernel.

Thank you,
 Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: Re: megaraid_mbox: garbage in file
Posted by vaverin on Fri, 12 May 2006 04:15:46 GMT
View Forum Message <> Reply to Message

Vasily Averin wrote:
> Ju, Seokmann wrote:
>>I'm waiting for feedback from F/W team for MegaRAID 150-4 controller if it supports 64-bit DMA.
>>
>>I'll update here as I get.

Could you please tell me any updates? Could you confirm that this issue was
reproduced on your nodes?

>>Can you do one quick change in the driver?
>>Search for 'pci_set_dma_mask()' API calls in the driver and mask out one of them with DMA_64BIT_MASK as follow.
>>---
>> // if (pci_set_dma_mask(adapter->pdev, DMA_64BIT_MASK) != 0) {
>>
>> // conlog(CL_ANN, (KERN_WARNING
>> //   "megaraid: could not set DMA mask for 64-bit.\n"));
>>
>> // goto out_free_sysfs_res;
>> // }
>>---
>>
>>I found that the driver is NOT checking 64-bit DMA capability of the controllers accordingly and this could be a reason.
>
> This change help me:
> Errors go away, file content is correct.

I'm going to use this change in production, at least as temporal workaround.
Could you please confirm that it is safe for all controllers supported by this
driver?

Thank you,
 Vasily Averin

SWsoft Virtuozzo/OpenVZ Linux kernel team

Subject: RE: megaraid_mbox: garbage in file
Posted by Seokmann.Ju on Fri, 12 May 2006 12:19:48 GMT
View Forum Message <> Reply to Message

Hi,

Friday, May 12, 2006 12:19 AM, Vasily Averin wrote:
> Could you please tell me any updates? Could you confirm that
> this issue was
> reproduced on your nodes?
Yes, it has confirmed by F/W team that the controller doesn't support 64-bit DMA.
A patch addresses the issue will followed by soon.

Thank you,

> -----Original Message-----
> From: Vasily Averin [mailto:vvs@sw.ru]
> Sent: Friday, May 12, 2006 12:19 AM
> To: Vasily Averin
> Cc: Ju, Seokmann; James Bottomley;
> linux-scsi@vger.kernel.org; Kolli, Neela; Mukker, Atul;
> Bagalkote, Sreenivas; devel@openvz.org; Linux Kernel Mailing List
> Subject: Re: megaraid_mbox: garbage in file
>
> Vasily Averin wrote:
> > Ju, Seokmann wrote:
> >>I'm waiting for feedback from F/W team for MegaRAID 150-4
> controller if it supports 64-bit DMA.
> >>
> >>I'll update here as I get.
>
> Could you please tell me any updates? Could you confirm that
> this issue was
> reproduced on your nodes?
>
> >>Can you do one quick change in the driver?
> >>Search for 'pci_set_dma_mask()' API calls in the driver and
> mask out one of them with DMA_64BIT_MASK as follow.
> >>---
> >> // if (pci_set_dma_mask(adapter->pdev, DMA_64BIT_MASK) != 0) {
> >>
> >> // conlog(CL_ANN, (KERN_WARNING
> >> //   "megaraid: could not set DMA mask for
> 64-bit.\n"));
> >>
> >> // goto out_free_sysfs_res;
> >> // }
> >>---
> >>
> >>I found that the driver is NOT checking 64-bit DMA
> capability of the controllers accordingly and this could be a reason.
> >
> > This change help me:
> > Errors go away, file content is correct.

>
> I'm going to use this change in production, at least as
> temporal workaround.
> Could you please confirm that it is safe for all controllers
> supported by this
> driver?
>
> Thank you,
> Vasily Averin
>
> SWsoft Virtuozzo/OpenVZ Linux kernel team
>