rbd - Bug #7790

Kernel panic when creating ZFS pools on CEPH RBD devices

03/19/2014 03:37 PM - Chris Dunlop

Status:	Resolved	**Start date:**	03/19/2014
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour
Source:	Support	**Reviewed:**	
Tags:		**Affected Versions:**	
Backport:		**ceph-qa-suite:**	
Regression:	No	**Pull request ID:**	
Severity:	3 - minor	**Crash signature:**	

Description

Creating a ZFS pool on top of krbd causes a kernel panic.

From the ZFSonLinux bug tracker (https://github.com/zfsonlinux/spl/issues/241):

"It seems the rbd driver will overwrite memory in some cases when passed odd-sized bio requests from ZFS. I gave up trying to track down the memory corruption because I couldn't get anything reproducible so I compared ZFS' block I/O to that of other file systems (ext4 and xfs) and discovered that the others only presented nice evenly-sized requests (usually 8 sectors or multiples of 8) but that ZFS presents "unusual" sizes such as 5, 13 and 15 sectors. Apparently the rbd driver doesn't like some aspect of these requests and the result is memory corruption. I've not yet determined whether the transfer size or the buffer alignment is the problem."

History

#1 - 03/20/2014 11:17 AM - Sheldon Mustard

- Source changed from other to Support

updated source to support.

#2 - 03/25/2014 06:25 AM - Andrea Ieri

Steps to reproduce bug:

- create a zpool with default options (zpool create mypool /dev/rbd/rbdpool/rbddev)
- wait a few seconds, sometimes a minute

Workaround:

use ashift=12 (zpool create -o ashift=12 mypool /dev/rbd/rbdpool/rbddev)

Versions under test:

```
[root@lxbse14c09 ~]# uname -r
3.13.7-1.el6.elrepo.x86_64
[root@lxbse14c09 ~]# rpm -q zfs spl
zfs-0.6.2-177_g98fad86.el6.x86_64
spl-0.6.2-23_g4c99541.el6.x86_64
```

Sample log from /var/log/messages:

```
[root@lxbse14c09 ~]# grep '14:03:51' /var/log/messages
Mar 25 14:03:51 lxbse14c09 kernel: general protection fault: 0000 [#1] SMP
Mar 25 14:03:51 lxbse14c09 kernel: Modules linked in: cbc rbd libceph libcrc32c zfs(PO) zcommon(PO) znvpair(PO
```

```
) zavl(PO) zunicode(PO) spl(O) autofs4 lockd sunrpc ipv6 iTCO_wdt iTCO_vendor_support microcode pcspkr sb_edac
 edac_core joydev lpc_ich sg igb hwmon ptp pps_core i2c_i801 ioatdma dca ext4 jbd2 mbcache sd_mod crc_t10dif c
rct10dif_common ahci libahci isci libsas scsi_transport_sas wmi mgag200 ttm drm_kms_helper sysimgblt sysfillre
ct syscopyarea dm_mirror dm_region_hash dm_log dm_mod
Mar 25 14:03:51 lxbse14c09 kernel: CPU: 5 PID: 4173 Comm: spl_kmem_cache/ Tainted: P          IO 3.13.7-1.el6.
elrepo.x86_64 #1
Mar 25 14:03:51 lxbse14c09 kernel: Hardware name: Intel Corporation S2600JF/S2600JF, BIOS SE5C600.86B.01.06.00
02.110120121539 11/01/2012
Mar 25 14:03:51 lxbse14c09 kernel: task: ffff880c1bdcb2d0 ti: ffff880c1bdfc000 task.ti: ffff880c1bdfc000
Mar 25 14:03:51 lxbse14c09 kernel: RIP: 0010:[<ffffffff81149e79>]  [<ffffffff81149e79>] get_page_from_freelist
+0x439/0x770
Mar 25 14:03:51 lxbse14c09 kernel: RSP: 0018:ffff880c1bdfda38  EFLAGS: 00010086
Mar 25 14:03:51 lxbse14c09 kernel: RAX: dead000000200200 RBX: 0000000000000010 RCX: 0000000000000000
Mar 25 14:03:51 lxbse14c09 kernel: RDX: dead000000100100 RSI: 0000000000000000 RDI: 0000000000000000
Mar 25 14:03:51 lxbse14c09 kernel: RBP: ffff880c1bdfdb38 R08: 0000000000000000 R09: 000000000050b380
Mar 25 14:03:51 lxbse14c09 kernel: R10: 0000000000003026 R11: 0000000000000000 R12: ffff88063f6b7360
Mar 25 14:03:51 lxbse14c09 kernel: R13: ffff88063ffd9d80 R14: 0000000000000000 R15: ffffea0015256e90
Mar 25 14:03:51 lxbse14c09 kernel: FS:  0000000000000000(0000) GS:ffff88063f6a0000(0000) knlGS:000000000000000
0
Mar 25 14:03:51 lxbse14c09 kernel: CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
Mar 25 14:03:51 lxbse14c09 kernel: CR2: 0000003d2da48123 CR3: 0000000001c0c000 CR4: 00000000000407e0
Mar 25 14:03:51 lxbse14c09 kernel: Stack:
Mar 25 14:03:51 lxbse14c09 kernel: ffff88061fcbc800 0000000000000000 ffff880c1bdcbd98 ffff88061fcbc000
Mar 25 14:03:51 lxbse14c09 kernel: ffff880c1bdfda98 ffffffff810a076c 0000000000000000 0000000100000000
Mar 25 14:03:51 lxbse14c09 kernel: ffff88063ffdab08 0000000000000002 ffff88063ffd9de8 0000000000000000
Mar 25 14:03:51 lxbse14c09 kernel: Call Trace:
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff810a076c>] ? update_group_power+0x2c/0x150
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8114ca72>] __alloc_pages_nodemask+0x152/0x330
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8118f142>] alloc_pages_current+0xb2/0x170
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8117e6e2>] __vmalloc_area_node+0x112/0x1d0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8117e826>] __vmalloc_node_range+0x86/0xd0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa0317081>] ? kv_alloc+0x51/0x60 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8117e8a5>] __vmalloc_node+0x35/0x40
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa0317081>] ? kv_alloc+0x51/0x60 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8117ea32>] __vmalloc+0x22/0x30
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa0317081>] kv_alloc+0x51/0x60 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa03170ba>] spl_slab_alloc+0x2a/0x3a0 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff81617fd8>] ? __schedule+0x3b8/0x6b0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa0317460>] spl_cache_grow_work+0x30/0xd0 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa031b45d>] taskq_thread+0x22d/0x4f0 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8109c2b0>] ? try_to_wake_up+0x2c0/0x2c0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffffa031b230>] ? task_expire+0x120/0x120 [spl]
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8108bb6e>] kthread+0xce/0xf0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8108baa0>] ? kthread_freezable_should_stop+0x70/0x70
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff816243fc>] ret_from_fork+0x7c/0xb0
Mar 25 14:03:51 lxbse14c09 kernel: [<ffffffff8108baa0>] ? kthread_freezable_should_stop+0x70/0x70
Mar 25 14:03:51 lxbse14c09 kernel: Code: 04 00 48 89 45 88 89 95 38 ff ff ff 89 4d 84 e9 b3 00 00 00 66 0f 1f
44 00 00 4d 8b 7c 1c 08 49 83 ef 20 49 8b 57 20 49 8b 47 28 <48> 89 42 08 48 89 10 48 ba 00 01 10 00 00 00 ad
de 48 b8 00 02
Mar 25 14:03:51 lxbse14c09 kernel: RIP  [<ffffffff81149e79>] get_page_from_freelist+0x439/0x770
Mar 25 14:03:51 lxbse14c09 kernel: RSP <ffff880c1bdfda38>
Mar 25 14:03:51 lxbse14c09 kernel: ---[ end trace ca8a7b46dbce99bb ]---
```

See also the linked bug on the ZoL tracker for further details.

#3 - 04/08/2014 01:42 PM - Ian Colle

- Project changed from Ceph to rbd

#4 - 05/14/2014 01:49 AM - Ilya Dryomov

- Status changed from New to Resolved

```
libceph: fix corruption when using page_count 0 page in rbd
```

in the testing branch.