Ceph - Bug #10985

Some OSDs don't get up after upgrade from v0.92 to v0.93

03/02/2015 10:59 AM - Anonymous

Status:	Won't Fix	**% Done:**	0%
Priority:	Urgent	**Spent time:**	0.00 hour
Assignee:	Loic Dachary		
Category:			
Target version:	v0.94		
Source:	Community (dev)	**Reviewed:**	
Tags:		**Affected Versions:**	v0.93 - Last Hammer Sprint
Backport:		**ceph-qa-suite:**	
Regression:	No	**Pull request ID:**	
Severity:	3 - minor	**Crash signature:**	

Description

Workaround

- get a ceph-osd binary from v0.92
- ceph-osd -i X --flush-journal
- restart the OSD

Updated release notes : f2b3192f3629f669c6ed4f1e0ad230069a90ae28

Description

Hello,

after upgrading my ceph cluster from 0.92 to 0.93 earlyer this morning 10 OSDs out of 24 are down.

```
-2> 2015-03-02 09:37:15.734525 7f128a0bd880  3 journal journal_replay: applying op seq 3629506
    -1> 2015-03-02 09:37:15.734706 7f128a0bd880 10 journal op_apply_start 3629506 open_ops 0 -> 1
     0> 2015-03-02 09:37:15.737648 7f128a0bd880 -1 os/Transaction.cc: In function 'void ObjectStor
e::Transaction::_build_actions_from_tbl()' thread 7f128a0bd880 time 2015-03-02 09:37:15.734724
os/Transaction.cc: 504: FAILED assert(ops == data.ops)
```

```
ceph version 0.93 (bebf8e9a830d998eeaab55f86bb256d4360dd3c4)
 1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x85) [0xbc7e75]
 2: (ObjectStore::Transaction::_build_actions_from_tbl()+0x3476) [0x9b2156]
 3: (FileStore::_do_transaction(ObjectStore::Transaction&, unsigned long, int, ThreadPool::TPHandl
e*)+0x3af0) [0x9359e0]
 4: (FileStore::_do_transactions(std::list&lt;ObjectStore::Transaction*, std::allocator&lt;ObjectS
tore::Transaction*&gt; >&, unsigned long, ThreadPool::TPHandle*)+0x64) [0x937954]
 5: (JournalingObjectStore::journal_replay(unsigned long)+0x5db) [0x9500bb]
 6: (FileStore::mount()+0x3730) [0x922180]
 7: (OSD::init()+0x26c) [0x6b828c]
 8: (main()+0x27f3) [0x6438e3]
 9: (__libc_start_main()+0xf5) [0x7f128745faf5]
 10: /usr/bin/ceph-osd() [0x65c849]
 NOTE: a copy of the executable, or `objdump -rdS &lt;executable&gt;` is needed to interpret this.
```

Adding tarball containing:
- backtrace - file "osd.3.backtrace" (just for osd.3, but others produce same error)
- "ceph report" report - file "ceph_report"
- log files of failed OSDs - directory logs_of_failed_osds

Related issues:		
Related to Ceph - Bug #10734: v0.91 to v0.92 upgrade journal replay crash	**Resolved**	**02/03/2015**
Duplicated by Ceph - Bug #10998: Multiple OSDs are down and will not restart:...	**Duplicate**	**03/03/2015**

History

#1 - 03/02/2015 11:02 AM - Loic Dachary

- Status changed from New to 12

- Priority changed from High to Urgent

could it be related to https://ceph.com/git/?p=ceph.git;a=commit;h=2598fc50749f7a1e8450c07b27cec5fa54b3e41f ?

#2 - 03/02/2015 01:10 PM - Loic Dachary

- Target version changed from v0.93 - Last Hammer Sprint to v0.94

#3 - 03/02/2015 05:20 PM - Loic Dachary

- File journal.dump added

attaching the output of ceph-osd dump from a journal provided by Edgaras Lukosevicius

#4 - 03/02/2015 05:35 PM - Loic Dachary

```
#5  0x00000031aa25ef93 in __cxa_throw () from /lib64/libstdc++.so.6
#6  0x00000000019f5b5c in ceph::__ceph_assert_fail (assertion=0x1bdf07d "ops == data.ops",
    file=0x1bdef4f "os/Transaction.cc", line=504,
    func=0x1bdf540 <ObjectStore::Transaction::_build_actions_from_tbl()::__PRETTY_FUNCTION__> "void ObjectStor
e::Transaction::_build_actions_from_tbl()") at common/assert.cc:77
#7  0x00000000017da904 in ObjectStore::Transaction::_build_actions_from_tbl (this=0x49608c0) at os/Transaction
.cc:504
#8  0x00000000017564be in ObjectStore::Transaction::begin (this=0x49608c0) at os/ObjectStore.h:856
#9  0x00000000017db36d in ObjectStore::Transaction::dump (this=0x49608c0, f=0x7fffffffccf0) at os/Transaction.
cc:513
#10 0x0000000000185a407 in FileJournal::dump (this=0x4958000, out=...) at os/FileJournal.cc:612
#11 0x0000000000172281e in FileStore::dump_journal (this=0x4920000, out=...) at os/FileStore.cc:681
#12 0x0000000001343205 in main (argc=4, argv=0x7fffffffe218) at ceph_osd.cc:280
```

#5 - 03/02/2015 06:31 PM - Loic Dachary

the first transaction of the journal is struct_v == 8 and decode8_5 does not throw therefore use_tbl = true although it is not how the transaction was encoded.

#6 - 03/02/2015 10:35 PM - Loic Dachary

Binary data of the entry being decoded:

```
(gdb) x/16 bl._buffers.front()._raw.data
0x48f0800:      0x03c30508      0x030c0000      0x00030000      0x00080000
0x48f0810:      0x625f0000      0x6e696769      0x00086f66      0x00000000
```

```
0x48f0820:      0x00000000      0x00060000      0x655f0000      0x68636f70
0x48f0830:      0x00000004      0x00001249      0x00000005      0x666e695f
(gdb) bt
#0  FileJournal::read_entry (this=0x4960000, bl=..., next_seq=@0x7fffffffcfa0: 0, corrupt=0x0) at os/FileJourn
al.cc:1764
#1  0x0000000001867195 in FileJournal::read_entry (this=0x4960000, bl=..., last_seq=@0x7fffffffcfa0: 0) at os/
FileJournal.h:460
#2  0x000000000185a229 in FileJournal::dump (this=0x4960000, out=...) at os/FileJournal.cc:597
#3  0x000000000172281e in FileStore::dump_journal (this=0x4928000, out=...) at os/FileStore.cc:681
#4  0x0000000001343205 in main (argc=4, argv=0x7fffffffe1d8) at ceph_osd.cc:280
```

#7 - 03/02/2015 10:39 PM - Loic Dachary

After decoding by decode8_5

```
print/x data
$6 = {ops = 0x30000030c, largest_data_len = 0x6f666e69, largest_data_off = 0x8, largest_data_off_in_tbl = 0x0,
 fadvise_flags = 0x5f000000}
bt
#0  ObjectStore::Transaction::decode8_5 (this=0x49688c0, bl=..., struct_v=8 '\b') at ./os/ObjectStore.h:1644
#1  0x0000000001693478 in ObjectStore::Transaction::decode (this=0x49688c0, bl=...) at ./os/ObjectStore.h:1588
#2  0x000000000177443c in ObjectStore::Transaction::Transaction (this=0x49688c0, dp=...) at os/ObjectStore.h:1
534
#3  0x000000000185a3c4 in FileJournal::dump (this=0x4960000, out=...) at os/FileJournal.cc:609
#4  0x000000000172281e in FileStore::dump_journal (this=0x4928000, out=...) at os/FileStore.cc:681
#5  0x0000000001343205 in main (argc=4, argv=0x7fffffffe1d8) at ceph_osd.cc:280
```

#8 - 03/02/2015 11:04 PM - Loic Dachary

- Description updated

#9 - 03/02/2015 11:12 PM - Loic Dachary

When decoded with !use_tbl, the data structure is:

```
$1 = {ops = 1, largest_data_len = 0, largest_data_off = 0, largest_data_off_in_tbl = 0, fadvise_flags = 0}
```

#10 - 03/02/2015 11:16 PM - Loic Dachary

- Description updated

#11 - 03/03/2015 07:59 AM - Kefu Chai

```
(gdb) x/16 bl._buffers.front()._raw.data
0x48f0800:      0x03c30508      0x030c0000      0x00030000      0x00080000
0x48f0810:      0x625f0000      0x6e696769      0x00086f66      0x00000000
0x48f0820:      0x00000000      0x00060000      0x655f0000      0x68636f70
0x48f0830:      0x00000004      0x00001249      0x00000005      0x666e695f
```

so this buffer is a valid encoded transaction from the POV of decode8_5(), where the 0x030c is taken as data.ops, but it should be decoded as data_bl.size(). and we don't have enough bits for verification other than the struct_compat/struct_v and size of buffer. if the bl is large enough we can hardly tell a version 9 encoded transaction who claims to be version 8 from the one in real version 8.

maybe we can also check the fadvise_flags with CEPH_OSD_OP_FLAG_FADVISE_DONTNEED ? i double checked the source of v0.92, the only place where we update the TransactionData.advise_flags is ECBackend::handle_sub_write(). in this method, we set the advise_flags to CEPH_OSD_OP_FLAG_FADVISE_DONTNEED at ECBackend.cc:841 .

so any encoded transaction whose advise_flags is not 0 or CEPH_OSD_OP_FLAG_FADVISE_DONTNEED, must be a bad one (or we should try to decode it in another way). i understand there is still chance of false negative. but might help in some cases if user fails to read https://github.com/ceph/ceph/commit/f2b3192f3629f669c6ed4f1e0ad230069a90ae28 ?

#12 - 03/03/2015 05:54 PM - Loic Dachary

- Description updated

- Status changed from 12 to Won't Fix

- Assignee set to Loic Dachary

Files

ceph_upgrade_v092_to_v093__half_osds_down.tar.gz	255 KB	03/02/2015	Anonymous
journal.dump	13.3 KB	03/02/2015	Loic Dachary