

RADOS - Bug #37840

FAILED assert(0 == "we got a bad state machine event") after upgrade from 13.2.2 to 13.2.4

01/09/2019 12:46 PM - Alec Blayne

Status:	New	Start date:	01/09/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	v13.2.4	Spent time:	0.00 hour
Source:		Affected Versions:	
Tags:		ceph-qa-suite:	
Backport:		Component(RADOS):	
Regression:	Yes	Pull request ID:	
Severity:	2 - major	Crash signature:	
Reviewed:			

Description

Running a 3 node cluster, no issues on two of the hosts, but one of the hosts has osds crashing like this:

--- begin dump of recent events ---

```
10> 2019-01-09 12:25:42.128 7f2d07a15700-5- 10.5.2.101:6801/21360 >> 10.5.2.101:6805/35561 conn(0x55e400184000 :-1
s=STATE_OPEN_MESSAGE_READ_FOOTER_AND_DISPATCH pgs=6 cs=1 l=0). rx osd.3 seq 39 0x55e3f5f13800 PGlog(24.1d
log log((43440'8144340,43440'8147389), crt=43440'8147389) pi ([0,0] intervals=) e43768/43768) v5
9> 2019-01-09 12:25:42.128 7f2d07a15700-1- 10.5.2.101:6801/21360 <== osd.3 10.5.2.101:6805/35561 39 ==== PGlog(24.1d log
log((43440'8144340,43440'8147389), crt=43440'8147389) pi ([0,0] intervals=) e43768/43768) v5 ==== 360920+0+0 (4130530693 0
0) 0x55e3f5f13800 con 0x55e400184000
-8> 2019-01-09 12:25:42.128 7f2cee7d4700 -1 /var/tmp/portage/sys-cluster/ceph-13.2.4/work/ceph-13.2.4/src/osd/PG.cc: In function
'PG::RecoveryState::Crashed::Crashed(boost::statechart::state<PG::RecoveryState::Crashed,
PG::RecoveryState::RecoveryMachine>::my_context)' thread 7f2cee7d4700 time 2019-01-09 12:25:42.126930
/var/tmp/portage/sys-cluster/ceph-13.2.4/work/ceph-13.2.4/src/osd/PG.cc: 6607: FAILED assert(0 == "we got a bad state machine
event")
```

ceph version 13.2.4 (b10be4d44915a4d78a8e06aa31919e74927b142e) mimic (stable)

```
1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x102) [0x7f2d0b94e5f2]
2: (()+0x2a0787) [0x7f2d0b94e787]
3: (PG::RecoveryState::Crashed::Crashed(boost::statechart::state<PG::RecoveryState::Crashed, P
G::RecoveryState::RecoveryMachine, boost::mpl::list<mpl::na, mpl::na, mpl::na, mpl::na, mpl
::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, m
pl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na>&;, (boost::statechart::history_mode)0>
::my_context)+0x88) [0x55e3f32ba3a8]
4: (boost::statechart::state<PG::RecoveryState::Crashed, PG::RecoveryState::RecoveryMachine, b
oost::mpl::list<mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na,
mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::n
a, mpl::na, mpl::na>&;, (boost::statechart::history_mode)0>::deep_construct(boost::statechart::
state_machine<PG::RecoveryState::RecoveryMachine, PG::RecoveryState::Initial, std::allocator<
;void>&;, boost::statechart::null_exception_translator>* const&, boost::statechart::state_machine
<PG::RecoveryState::RecoveryMachine, PG::RecoveryState::Initial, std::allocator<void>&;, bo
ost::statechart::null_exception_translator>&)+0x36) [0x55e3f332e496]
5: (boost::statechart::simple_state<PG::RecoveryState::Peering, PG::RecoveryState::Primary, PG
::RecoveryState::GetInfo, (boost::statechart::history_mode)0>&;::react_impl(boost::statechart::ev
ent_base const&, void const*)+0x2b4) [0x55e3f3331d84]
6: (boost::statechart::simple_state<PG::RecoveryState::GetMissing, PG::RecoveryState::Peering,
boost::mpl::list<mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::n
a, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::na, mpl::
na, mpl::na, mpl::na>&;, (boost::statechart::history_mode)0>::react_impl(boost::statechart::ev
ent_base const&, void const*)+0x52) [0x55e3f3322c22]
7: (PG::do_peering_event(std::shared_ptr<PGPeeringEvent>&;, PG::RecoveryCtx*)+0x2c7) [0x55e3f
32ece77]
8: (OSD::dequeue_peering_evt(OSDShard*, PG*, std::shared_ptr<PGPeeringEvent>&;, ThreadPool::T
```

```
PHandle&)+0xf0) [0x55e3f323e510]
 9: (PGPeeringItem::run(OSD*, OSDShard*, boost::intrusive_ptr<PG>&&, ThreadPool::TPHandle&)+0
x52) [0x55e3f3484602]
10: (OSD::ShardedOpWQ::_process(unsigned int, ceph::heartbeat_handle_d*)+0x81d) [0x55e3f323f74d]
11: (ShardedThreadPool::shardedthreadpool_worker(unsigned int)+0x3d6) [0x7f2d0b953156]
12: (ShardedThreadPool::WorkThreadSharded::entry()+0x10) [0x7f2d0b954800]
13: (()+0x83f3) [0x7f2d0ae283f3]
14: (clone()+0x3f) [0x7f2d0a98052f]
NOTE: a copy of the executable, or `objdump -rdS &lt;executable>` is needed to interpret this.
```

And if I try to run the osd daemons, monitor also crashes with:

```
68 objects degraded (47.916%), 146 pgs degraded, 146 pgs undersized
-2> 2019-01-09 12:24:53.014 7f885681f700 10 maybe_remove_pg_upmaps
-1> 2019-01-09 12:24:53.014 7f885681f700 10 clean_pg_upmaps
0> 2019-01-09 12:24:53.014 7f885681f700 -1 ** Caught signal (Segmentation fault) *
in thread 7f885681f700 thread_name:safe_timer

ceph version 13.2.4 (b10be4d44915a4d78a8e06aa31919e74927b142e) mimic (stable)
 1: (()+0x4eb6f3) [0x55a190d296f3]
 2: (()+0x147e0) [0x7f885d7737e0]
 3: (OSDMap::check_health(health_check_map_t*) const+0x52e) [0x7f885e07195e]
 4: (OSDMonitor::encode_pending(std::shared_ptr<MonitorDBStore::Transaction>)+0x30d5) [0x55a
190c96bf5]
 5: (PaxosService::propose_pending()+0x1e6) [0x55a190c4fcf6]
 6: (C_MonContext::finish(int)+0x39) [0x55a190afd239]
 7: (Context::complete(int)+0x9) [0x55a190b30749]
 8: (SafeTimer::timer_thread()+0xf9) [0x7f885df17da9]
 9: (SafeTimerThread::entry()+0xd) [0x7f885df194cd]
10: (()+0x83f3) [0x7f885d7673f3]
11: (clone()+0x3f) [0x7f885d2c552f]
NOTE: a copy of the executable, or `objdump -rdS &lt;executable>` is needed to interpret this.
```

History

#1 - 01/21/2019 10:23 PM - Greg Farnum

- Project changed from Ceph to RADOS