RADOS - Bug #38784

osd: FAILED ceph_assert(attrs || !pg_log.get_missing().is_missing(soid) || (it_objects != pg_log.get_log().objects.end() && &it_objects->second->op == pg_log_entry_t::LOST_REVERT)) in PrimaryLogPG::get_object_context()

03/15/2019 06:06 PM - Neha Ojha

Status: Resolved
Priority: Normal
Assignee: Neha Ojha
Category: Target version:
Source: Affected Versions: Tags: Backport: luminous,mimic,nautilus Regression: No Severity: 3 - minor Reviewed: 

Description

2019-03-14T01:34:27.455 INFO:tasks.ceph.osd.3.smithi131.stderr:/home/jenkins-build/build/workspace/ceph-dev-new-build/ARCH/x86_64/AVAILABLE_ARCH/x86_64/AVAILABLE_DIST/centos7/DIST/centos7/MACHINE_SIZE/huge/release/14.0.1-3842-g08f436c/rpm/el7/BUILD/ceph/14.0.1-3842-g08f436c/src/osd/PrimaryLogPG.cc: In function 'ObjectContextRef PrimaryLogPG::get_object_context(const hobject_t&, bool, const std::map<std::basic_string<char>, ceph::buffer::list>*) thread 7fe774509700 time 2019-03-14 01:34:27.471330

2019-03-14T01:34:27.456 INFO:tasks.ceph.osd.3.smithi131.stderr:/home/jenkins-build/build/workspace/ceph-dev-new-build/ARCH/x86_64/AVAILABLE_ARCH/x86_64/AVAILABLE_DIST/centos7/DIST/centos7/MACHINE_SIZE/huge/release/14.0.1-3842-g08f436c/rpm/el7/BUILD/ceph/14.0.1-3842-g08f436c/src/osd/PrimaryLogPG.cc: 10998: FAILED ceph_assert(attrs || !pg_log.get_missing().is_missing(soid) || (it_objects != pg_log.get_log().objects.end() && &it_objects->second->op == pg_log_entry_t::LOST_REVERT))

2019-03-14T01:34:27.456 INFO:tasks.ceph.osd.3.smithi131.stderr: ceph version 14.0.1-3842-g08f436c (08f436c23590d7ac8ad260a72c6b942440ace5a6) nautilus (dev)

2019-03-14T01:34:27.456 INFO:tasks.ceph.osd.3.smithi131.stderr: 1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x14a) [0x55a3af940acc]

2019-03-14T01:34:27.456 INFO:tasks.ceph.osd.3.smithi131.stderr: 2: (ceph::__ceph_assertf_fail(char const*, char const*, int, char const*, char const*, ...)+0) [0x55a3af940c9a]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 3: (PrimaryLogPG::get_object_context(hobject_t const&, bool, std::map<std::string, ceph::buffer::list, std::less<std::string>, std::allocator<std::pair<std::string const, ceph::buffer::list> > > const*) thread 7fe774509700 time 2019-03-14 01:34:27.471330

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 4: (PrimaryLogPG::prepare_object_replay_deletes(hobject_t const&, eversion_t, PGBackend::RecoveryHandle*, bool*)+0x84) [0x55a3afbf6934]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 5: (PrimaryLogPG::recover_replicas(unsigned long, ThreadPool::TPHandle*, bool*)+0xb74) [0x55a3afbf8e94]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 6: (PrimaryLogPG::start_recovery_objects(unsigned long, ThreadPool::TPHandle*, unsigned long*)+0x1bd) [0x55a3afcf4ebe]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 7: (OSD::do_recovery(PG*, unsign int, unsigned long, ThreadPool::TPHandle&)+0x363) [0x55a3afa7ca3e]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 8: (PGRecovery::run(OSD*, OSDShard *, boost::intrusive_ptr<PG>*, ThreadPool::TPHandle*)+0x19) [0x55a3afdf38f]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 9: (OSD::ShardedOpWQ::process(unsigned int, ceph::heartbeat_handle_d*)+0x4ac) [0x55a3af9af1ac]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 10: (ShardedThreadPool::shardedthreadpool_worker(unsigned int)+0x433) [0x55a3b008c803]

2019-03-14T01:34:27.457 INFO:tasks.ceph.osd.3.smithi131.stderr: 11: (ShardedThreadPool::WorkThreadSharded::entry()+0x10) [0x55a3b008f8a0]

2019-03-14T01:34:27.458 INFO:tasks.ceph.osd.3.smithi131.stderr: 12: (())+0x7dd5) [0x7fe797dd2cdd5]

2019-03-14T01:34:27.458 INFO:tasks.ceph.osd.3.smithi131.stderr: 13: (clone()+0x6d) [0x7fe79cbf2ead
The problem seems to be that on the primary-osd.3, we have 1 missing and 1 unfound object, which are not the same object. In start_recovery_ops(), since num_missing == num_unfound, we end up calling recover_replicas(), before recovering the missing object on the primary. This ends up with `pg_log.get_missing().is_missing(soid)` not being true during recovery of the missing object on a replica.

We should account for the unfound object in the missing set of the primary, that way we would have had num_missing=2, and would have recovered the object that is only missing(not unfound) on the primary before recovering it on replicas.

**Related issues:**
- Copied to RADOS - Backport #39218: luminous: osd: FAILED ceph_assert(attrs ||... Resolved
- Copied to RADOS - Backport #39219: nautilus: osd: FAILED ceph_assert(attrs ||... Resolved
- Copied to RADOS - Backport #39220: mimic: osd: FAILED ceph_assert(attrs || lp... Resolved

**History**

#1 - 03/27/2019 04:38 PM - Neha Ojha

- Status changed from New to Fix Under Review
- Backport set to luminous,mimic,nautilus
- Pull request ID set to 27205

#2 - 03/30/2019 08:46 AM - xie xingguo

- Status changed from Fix Under Review to Pending Backport

#3 - 04/10/2019 09:07 PM - Nathan Cutler

- Copied to Backport #39218: luminous: osd: FAILED ceph_assert(attrs || `pg_log.get_missing().is_missing(soid) || (it_objects != pg_log.get_log().objects.end() && it_objects->second->op == pg_log_entry_t::LOST_REVERT)` in PrimaryLogPG::get_object_context() added

#4 - 04/10/2019 09:07 PM - Nathan Cutler

- Copied to Backport #39219: nautilus: osd: FAILED ceph_assert(attrs || `pg_log.get_missing().is_missing(soid) || (it_objects != pg_log.get_log().objects.end() && it_objects->second->op == pg_log_entry_t::LOST_REVERT)` in PrimaryLogPG::get_object_context() added

#5 - 04/10/2019 09:07 PM - Nathan Cutler

- Copied to Backport #39220: mimic: osd: FAILED ceph_assert(attrs || `pg_log.get_missing().is_missing(soid) || (it_objects != pg_log.get_log().objects.end() && it_objects->second->op == pg_log_entry_t::LOST_REVERT)` in PrimaryLogPG::get_object_context() added

#6 - 04/29/2019 06:29 AM - Prashant D

- Status changed from Pending Backport to In Progress
- Assignee set to Prashant D

#7 - 04/29/2019 11:18 PM - Prashant D

- Status changed from In Progress to Pending Backport
- Assignee changed from Prashant D to Neha Ojha

#8 - 06/20/2019 10:00 AM - Nathan Cutler

- Status changed from Pending Backport to Resolved