Ceph - Bug #11199
osd: ENOENT on clone

03/21/2015 10:23 PM - Sage Weil

Status: Resolved
Priority: High
Assignee: Samuel Just
Category:
Target version:
Source: Q/A
Backport: firefly
Regression:
Severity: 3 - minor

Description

2015-03-21 05:59:32.657469 7fb4bf854700 10 filestore(/var/lib/ceph/osd/ceph-4) error opening file /var/lib/ceph/osd/ceph-4/current/1.2_head/mira01213209-286___head_31BCCDB2___1 with flags=2: (2) No such file or directory
2015-03-21 05:59:32.657505 7fb4bf854700 0 filestore(/var/lib/ceph/osd/ceph-4) ENOENT on clone suggests osd bug

Revision 1388d6bd - 03/24/2015 10:29 PM - Samuel Just
Replicated PG: trim backfill intervals based on peer's last_backfill_started

Otherwise, we fail to trim the peer's last_backfill_started and get bug 11199.
1) osd 4 backfills up to 31bccdb2/mira01213209-286/head (henceforth: foo)
2) Interval change happens
3) osd 0 now finds itself backfilling to 4 (lb=foo) and osd.5 (lb=b6670ba2/mira01213209-160/snapdir//1, henceforth: bar)
4) recover_backfill causes both 4 and 5 to scan forward, so 4 has an interval starting at foo, 5 has an interval starting at bar.
5) Once those have come back, recover_backfill attempts to trim off the last_backfill_started, but 4's interval starts after that, so foo remains in osd 4's interval (this is the bug)
7) We serve a copyfrom on foo (sent to 4 as well).
8) We eventually get to foo in the backfilling. Normally, they would have the same version, but of course we don't update osd.4's interval from the log since it should not have received writes in that interval. Thus, we end up trying to recover foo on osd.4 anyway.
9) But, an interval change happens between removing foo from osd.4 and completing the recovery, leaving osd.4 without foo, but with lb >= foo

Fixes: #11199
ReplicatedPG: trim backfill intervals based on peer's last_backfill_started

Otherwise, we fail to trim the peer's last_backfill_started and get bug 11199.

1) osd 4 backfills up to 31bcbd2b/mira01213209-286/head (henceforth: foo)

2) Interval change happens

3) osd 0 now finds itself backfilling to 4 (lb=foo) and osd.5
   (lb=b6670b2e/mira01213209-160/snapdir//1, henceforth: bar)

4) recover_backfill causes both 4 and 5 to scan forward, so 4 has an interval
   starting at foo, 5 has an interval starting at bar.

5) Once those have come back, recover_backfill attempts to trim off the
   last_backfill_started, but 4's interval starts after that, so foo remains in
   osd 4's interval (this is the bug)

7) We serve a copyfrom on foo (sent to 4 as well).

8) We eventually get to foo in the backfilling. Normally, they would have the
   same version, but of course we don't update osd.4's interval from the log since
   it should not have received writes in that interval. Thus, we end up trying to
   recover foo on osd.4 anyway.

9) But, an interval change happens between removing foo from osd.4 and
   completing the recovery, leaving osd.4 without foo, but with lb >= foo

Fixes: #11199
Backport: firefly
Signed-off-by: Samuel Just <sjust@redhat.com>
(cherry picked from commit 1388d6bd949a18e8ac0aeb79fbb0316879)
This is kind of a fun one.

1) osd 4 backfills up to 31bccdb2/mira01213209-286/head (henceforth: foo)
2) Interval change happens
3) osd 0 now finds itself backfilling to 4 (lb=foo) and osd.5 (lb=b6670ba2/mira01213209-160/snapdir//1, henceforth: bar)
4) recover_backfill causes both 4 and 5 to scan forward, so 4 has an interval starting at foo, 5 has an interval starting at bar.
5) Once those have come back, recover_backfill attempts to trim off the last_backfill_started, but 4's interval starts after that, so foo remains in osd 4's interval (this is the bug)
6) We serve a copyfrom on foo (sent to 4 as well).
6) We eventually get to foo in the backfilling. Normally, they would have the same version, but of course we don't update osd.4's interval from the log since it should not have received writes in that interval. Thus, we end up trying to recover foo on osd.4 anyway.
7) But, an interval change happens between removing foo from osd.4 and completing the recovery, leaving osd.4 without foo, but with lb >= foo -->
this crash.

#3 - 03/24/2015 10:15 PM - Samuel Just
- Status changed from New to Testing

#4 - 03/27/2015 03:16 PM - Sage Weil
- Status changed from Testing to Pending Backport
- Priority changed from Urgent to High
- Backport set to firefly

#5 - 04/17/2015 09:31 AM - Xinxin Shu

- firefly backport [https://github.com/ceph/ceph/pull/4385](https://github.com/ceph/ceph/pull/4385)

#6 - 05/07/2015 11:41 AM - Loic Dachary

3fb97e2 ReplicatedPG: trim backfill intervals based on peer's last_backfill_started (in firefly),

#7 - 05/07/2015 11:41 AM - Loic Dachary

- Status changed from Pending Backport to Resolved