Ceph - Bug #11144

erasure-code-profile set races with erasure-code-profile rm

03/18/2015 12:53 PM - Loic Dachary

Status:	Resolved	**Start date:**	03/18/2015
Priority:	Urgent	**Due date:**	
Assignee:	Loic Dachary	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour
Source:	other	**Reviewed:**	
Tags:		**Affected Versions:**	
Backport:	firefly	**ceph-qa-suite:**	
Regression:	No	**Pull request ID:**	
Severity:	3 - minor		

Description

If erasure-code-profile set is called immediately after erasure-code-profile rm (i.e. before the pending osdmap was proposed) for the same profile, the profile will actually be deleted instead of being added (because removal happens after set when decoding an incremental change). In OSDMap.cc

```
  // erasure_code_profiles
  for (map<string,map<string,string> >::const_iterator i =
     inc.new_erasure_code_profiles.begin();
       i != inc.new_erasure_code_profiles.end();
       ++i) {
    set_erasure_code_profile(i->first, i->second);
  }

  for (vector<string>::const_iterator i = inc.old_erasure_code_profiles.begin();
       i != inc.old_erasure_code_profiles.end();
       ++i)
    erasure_code_profiles.erase(*i);
```

The erasure-code-profile set code should check if there is a pending removal and wait for the proposal to finish to avoid this race.

From /a/sage-2015-01-19_18:35:10-rados-wip-dho-distro-basic-multi/713735/remote/plana32/log/ceph-mon.b.log.gz

```
2015-01-19 22:08:05.466030 7f7ff6407700 10 mon.b@0(leader).osd e164 create_pending e 165
2015-01-19 22:08:05.466049 7f7ff6407700  1 -- 10.214.131.8:6789/0 --> 10.214.131.8:0/42028352 -- m
on_command_ack([{"prefix": "osd erasure-code-profile rm", "name": "testprofile"}]=0  v164) v1 -- ?
+0 0x2b11e00 con 0x2b53180
...
2015-01-19 22:08:05.492742 7f7ff6407700  7 mon.b@0(leader).osd e164 prepare_update mon_command({"p
refix": "osd erasure-code-profile set", "name": "testprofile", "profile": [ "k=2", "m=1", "ruleset
-failure-domain\
=osd"]} v 0) v1 from client.4287 10.214.131.8:0/43028352
2015-01-19 22:08:05.492852 7f7ff6407700 20 mon.b@0(leader).osd e164 erasure code profile testprofi
le set
2015-01-19 22:08:05.492863 7f7ff6407700 10 mon.b@0(leader).osd e164 should_propose
```

Related issues:

Related to Ceph - Bug #10488: osd erasure-code-profile set is sometimes not i...	**Resolved**	**01/08/2015**

Associated revisions

Revision 0d52aca0 - 03/18/2015 01:17 PM - Loic Dachary

osd: erasure-code-profile incremental rm before set

It is possible for an incremental change to have both a rm and a set for a given erasure code profile. It only happens when a rm is followed by a set. When a set is followed by a rm, the rm will remove the pending set in the incremental change.

The logic is the same for pool create and pool delete.

We must apply the incremental erasure-code-profile removal before the creation otherwise rm and set in the same proposal will ignore the set.

This fix is minimal. A better change would be that erasure-code-profile set checks if there is a pending removal and wait_for_finished_proposal before creating.

http://tracker.ceph.com/issues/11144 Fixes: #11144

Signed-off-by: Loic Dachary <ldachary@redhat.com>

Revision c0cfd6e5 - 04/09/2015 06:42 AM - Loic Dachary

osd: erasure-code-profile incremental rm before set

It is possible for an incremental change to have both a rm and a set for a given erasure code profile. It only happens when a rm is followed by a set. When a set is followed by a rm, the rm will remove the pending set in the incremental change.

The logic is the same for pool create and pool delete.

We must apply the incremental erasure-code-profile removal before the creation otherwise rm and set in the same proposal will ignore the set.

This fix is minimal. A better change would be that erasure-code-profile set checks if there is a pending removal and wait_for_finished_proposal before creating.

http://tracker.ceph.com/issues/11144 Fixes: #11144

Signed-off-by: Loic Dachary <ldachary@redhat.com>
(cherry picked from commit 0d52aca0d0c302983d03b0f5213ffed187e4ed63)

Conflicts:
src/osd/OSDMap.cc
resolved by replacing i++ with ++i

History

#1 - 03/18/2015 01:09 PM - Loic Dachary

The solution is for erasure-code-profile to remove from pending_inc, if any. In the same way pool delete does.

#2 - 03/18/2015 01:16 PM - Loic Dachary

Hum, not at all. pool delete + create do not have the same problem because the incremental change is applied the other way around : first delete then create. So if a delete + create are in the same proposal because the pool create does **not** cancel the removal, the pool will be deleted and created:

```
  for (set<int64_t>::const_iterator p = inc.old_pools.begin();
       p != inc.old_pools.end();
       ++p) {
    pools.erase(*p);
    name_pool.erase(pool_name[*p]);
    pool_name.erase(*p);
  }
  for (map<int64_t,pg_pool_t>::const_iterator p = inc.new_pools.begin();
       p != inc.new_pools.end();
       ++p) {
    pools[p->first] = p->second;
    pools[p->first].last_change = epoch;
  }
  for (map<int64_t,string>::const_iterator p = inc.new_pool_names.begin();
       p != inc.new_pool_names.end();
       ++p) {
    if (pool_name.count(p->first))
      name_pool.erase(pool_name[p->first]);
    pool_name[p->first] = p->second;
    name_pool[p->second] = p->first;
  }
```

#3 - 03/18/2015 01:29 PM - Loic Dachary

- Status changed from Verified to Need Review

https://github.com/ceph/ceph/pull/4066

#4 - 03/18/2015 06:53 PM - Loic Dachary

- Status changed from Need Review to Testing

#5 - 03/19/2015 03:54 PM - Samuel Just

- Status changed from Testing to Resolved

#6 - 03/19/2015 04:08 PM - Loic Dachary

- Status changed from Resolved to Pending Backport

- Backport set to firefly

#7 - 04/08/2015 04:50 AM - Xinxin Shu

firefly backport https://github.com/ceph/ceph/pull/4296

#8 - 04/17/2015 09:29 AM - Xinxin Shu

- firefly backport https://github.com/ceph/ceph/pull/4383

#9 - 05/05/2015 07:19 PM - Loic Dachary

- Status changed from Pending Backport to Resolved