MDS is crashed (mds/CDir.cc: 1391: FAILED assert(!is_complete()))

5/06/2015 10:28 AM - Andrey Matyashov

Status: Resolved
Priority: High
Assignee: Zheng Yan
Category: Other
Target version: Other
Source: Other
Tags: hammer
Backport: hammer
Regression: Yes
Severity: 1 - Critical

Description

After update from 0.82 to 0.94.1 mds crashed:

root@virt-master:~# /usr/bin/ceph-mds -i virt-master --debug_ms 9 --debug_mds 9 --pid-file /var/run/ceph/mds.virt-master.pid -c /etc/ceph/ceph.conf --cluster ceph -f > mds.virt-master.log

2015-05-06 13:13:07.076817 7f9e13a85780 -1 mds.-1.0 log_to_monitors {default=true} mds/CDir.cc: In function 'void CDir::fetch(MDSInternalContextBase*, const string&, bool)' thread 7f9e0a48f700 time 2015-05-06 13:13:11.230459

mds/CDir.cc: 1391: FAILED assert(!is_complete())

ceph version 0.94.1 (4ebfad3a3c51054df7e537a724c8d0b9be972ff)
1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x72) [0xa38d72]
2: (CDir::fetch(MDSInternalContextBase*, std::string const&, bool)+0x7fc) [0x86434c]
3: (CDir::fetch(MDSInternalContextBase*, bool)+0x2b) [0x8643db]
4: (MDCache::open_undef_inodes_dirfrags()+0x333) [0x74aec3]
5: (MDCache::rejoin_gather_finish()+0x333) [0x79745c]
6: (MDSInternalContextBase::complete(int)+0x15b) [0x8be9bb]
7: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::delete_me()+0x16) [0x6ba0b6]
8: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::sub_finish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]
9: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::delete_me()+0x16) [0x6c0ce2]
10: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::sub_finish(MDSInternalContextBase*, int)+0x9) [0x6af59]
11: (MDS::advance_queues()+0x112) [0x69e642]
12: (MDS::ProgressThread::entry()+0x4a) [0x69f12a]
13: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::delete_me()+0x16) [0x6ba0b6]
14: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::sub_finish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]

NOTE: a copy of the executable, or `objdump -rdS <executable>` is needed to interpret this.

2015-05-06 13:13:11.230253 7f9e0a48f700 -1 mds/CDir.cc: In function 'void CDir::fetch(MDSInternalContextBase*, const string&, bool)' thread 7f9e0a48f700 time 2015-05-06 13:13:11.230459

mds/CDir.cc: 1391: FAILED assert(!is_complete())

ceph version 0.94.1 (4ebfad3a3c51054df7e537a724c8d0b9be972ff)
1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x72) [0xa38d72]
2: (CDir::fetch(MDSInternalContextBase*, std::string const&, bool)+0x7fc) [0x86434c]
3: (CDir::fetch(MDSInternalContextBase*, bool)+0x2b) [0x8643db]
4: (MDCache::open_undef_inodes_dirfrags()+0x333) [0x74aec3]
5: (MDCache::rejoin_gather_finish()+0x333) [0x79745c]
6: (MDSInternalContextBase::complete(int)+0x15b) [0x8be9bb]
7: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::delete_me()+0x16) [0x6ba0b6]
8: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::subFinish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]
9: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::delete_me()+0x16) [0x6c0ce2]
10: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::subFinish(MDSInternalContextBase*, int)+0x9) [0x6af59]
13: (\(i\)+0x6b50) [0x7f9e13410b50]
14: (clone()\(i\)+0x8d) [0x7f9e1203095d]

NOTE: a copy of the executable, or `objdump -rdS <executable>` is needed to interpret this.

0> 2015-05-06 13:13:11.232253 7f9e0a48f700 -1 mds/CDir.cc: In function `void CDir::fetch(MDSInternalContextBase*, const
string&, bool)` thread 7f9e0a48f700 time 2015-05-06 13:13:11.230459

m/s/CDir.cc: 1391: FAILED assert(!is_complete())

ceph version 0.94.1 (e4bfad3a3c51054df7e537a724c8d0f9be972ff)
1: (ceph::__ceph_assert_fail(char const*, char const*, int, char const*)+0x72) [0x38d72]
2: (CDir::fetch(MDSInternalContextBase*, std::string const&, bool)+0x7fc) [0x86434c]
3: (CDir::fetch(MDSInternalContextBase*, bool)+0x2b) [0x8643db]
4: (MDCache::open_undef_inodes_dirfrags()+0x333) [0x74ace3]
5: (MDCache::rejoin_gather_finish()+0x6c) [0x79745c]

6: (MDSInternalContextBase::complete(int)+0x15b) [0x8be9bb]
7: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::delete_me()+0x16) [0x6ba0b6]
8: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::sub_finish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]
9: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub:::finish(int)+0x112) [0x6c0ce2]
10: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub::complete(int)+0x9) [0x6af59]
11: (MDS:::advance_queues()+0x112) [0x69e642]
12: (MDS::ProgressThread:::entry()+0x4a) [0x69f12a]
13: (\(i\)+0x6b50) [0x7f9e13410b50]
14: (clone()+0x6d) [0x7f9e1203095d]

NOTE: a copy of the executable, or `objdump -rdS <executable>` is needed to interpret this.

terminate called after throwing an instance of 'ceph::FailedAssertion'

*** Caught signal (Aborted) ***

in thread 7f9e0a48f700

ceph version 0.94.1 (e4bfad3a3c51054df7e537a724c8d0f9be972ff)
1: /usr/bin/ceph-mds() [0x9556ec]
2: (\(i\)+0x0a0) [0x7f9e13410b50]
3: (gsignal()+0x35) [0x7f9e11f87165]
4: (abort()+0x180) [0x7f9e11f8a3e0]
5: (\(_:\::__verbose_terminate_handler()+0x11d) [0x7f9e127db9c3]
6: (\(i\)+0x16) [0x7f9e1203095d]
7: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::delete_me()+0x16) [0x6ba0b6]
8: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::sub_finish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]
9: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub:::finish(int)+0x112) [0x6c0ce2]
10: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub::complete(int)+0x9) [0x6af59]
11: (MDS:::advance_queues()+0x112) [0x69e642]
12: (MDS::ProgressThread:::entry()+0x4a) [0x69f12a]
13: (\(i\)+0x6b50) [0x7f9e13410b50]
14: (clone()+0x6d) [0x7f9e1203095d]

2015-05-06 13:13:11.294984 7f9e0a48f700 -1 *** Caught signal (Aborted) ***

in thread 7f9e0a48f700

ceph version 0.94.1 (e4bfad3a3c51054df7e537a724c8d0f9be972ff)
1: /usr/bin/ceph-mds() [0x9556ec]
2: (\(i\)+0x0a0) [0x7f9e13410b50]
3: (gsignal()+0x35) [0x7f9e11f87165]
4: (abort()+0x180) [0x7f9e11f8a3e0]
5: (\(_:\::__verbose_terminate_handler()+0x11d) [0x7f9e127db9c3]
6: (\(i\)+0x16) [0x7f9e1203095d]
7: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::delete_me()+0x16) [0x6ba0b6]
8: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>:::sub_finish(MDSInternalContextBase*, int)+0x265) [0x6c0bd5]
9: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub:::finish(int)+0x112) [0x6c0ce2]
10: (C_GatherBase<MDSInternalContextBase, MDSInternalContextGather>::C_GatherSub::complete(int)+0x9) [0x6af59]
11: (MDS:::advance_queues()+0x112) [0x69e642]
12: (MDS::ProgressThread:::entry()+0x4a) [0x69f12a]
13: (\(i\)+0x6b50) [0x7f9e13410b50]
14: (clone()+0x6d) [0x7f9e1203095d]

2015-05-06 13:13:11.294984 7f9e0a48f700 -1 *** Caught signal (Aborted) ***

in thread 7f9e0a48f700
NOTE: a copy of the executable, or `objdump -rdS <executable>` is needed to interpret this.

Aborted

See more info in attached logfile. (start cmd: /usr/bin/ceph-mds -i virt-master --debug_ms 9 --debug_mds 9 -d --pid-file /var/run/ceph/mds.virt-master.pid -c /etc/ceph/ceph.conf --cluster ceph > mds.virt-master.log 2>&1 )

Related issues:

Copied to Backport #11737: MDS is crashed (mds/CDir.cc: 1391: FAILED assert(!... Resolved 05/06/2015

History

#1 - 05/06/2015 02:36 PM - John Spray

This may be due to the following change:

commit 818a80736c8b76c031f56708d03c263289686d51
Author: Yan, Zheng <zyan@redhat.com>
Date: Wed Dec 3 15:32:33 2014 +0800

  mds: drop dirty dentries in deleted directory

opened dirfrags and null dirty dentries in deleted directory inode
prevent MDCache::eval_stray() from purging the delete inode.

It's safe to not commit null dirty dentries in deleted directory to corresponding dirfrag objects, because these dirfrag objects will be deleted soon.

Fixes: #10164
Signed-off-by: Yan, Zheng <zyan@redhat.com>

#2 - 05/06/2015 02:50 PM - John Spray
The contradiction here seems to be that our code wants any unlinked directories (i.e. in a stray directory) to have no entries, but according to the contents of the cache there are entries, e.g. as a random example take:

dentry [dentry #100/stray4/10000093434/ActionTest.php [2,head] auth NULL (dversion lock) v=83 inode=0 | dirty=1 0x5239640]

I guess we need something during replay or rejoin to apply the try_remove_dentries_for_stray logic to the loaded directories, and clean out any of these dirty dentries before going further.

#3 - 05/06/2015 03:25 PM - John Spray
Created wip-11541-hammer-workaround branch. Andrey: once its built you should be able to find some packages on http://ceph.com/gitbuilder.cgi that you can try out (ask on IRC if unsure)

#4 - 05/06/2015 06:08 PM - Andrey Matyashov
Hi, i many restart mds daemons, and it started normally.

#5 - 05/08/2015 03:14 AM - Zheng Yan
I think the fix should be:

diff --git a/src/mds/CDir.cc b/src/mds/CDir.cc
index 211b8b0..23a2ff9 100644
--- a/src/mds/CDir.cc
+++ b/src/mds/CDir.cc
@@ -1414,9 +1414,17 @@ void CDir::fetch(MDSInternalContextBase *c, const string& want_dn, bool ignore_a
 // unlinked directory inode shouldn't have any entry
 if (inode->inode.nlink == 0 && !inode->snaprealm) {
     dout(7) << "fetch dirfrag for unlinked directory, mark complete" << dendl;
-    if (get_version() == 0)
+    if (get_version() == 0) {
+        if (state_test(STATE_REJOINUNDEF)) {
+            assert(cache->mds->is_rejoin());
+            state_clear(STATE_REJOINUNDEF);
+            cache->opened_undef_dirfrag(this);
+        } else {
-            set_version(1);
+            set_version(1);
+            if (get_version() == 0) {
+                assert(cache->mds->is_rejoin());
+                state_clear(STATE_REJOINUNDEF);
+                cache->opened_undef_dirfrag(this);
+            } else {
+                set_version(1);
+            }
+        }
+    }
+    if (get_version() == 0) {
+        if (state_test(STATE_REJOINUNDEF)) {
+            assert(cache->mds->is_rejoin());
+            state_clear(STATE_REJOINUNDEF);
+            cache->opened_undef_dirfrag(this);
+        } else {
+            set_version(1);
+        }
+        if (state_test(STATE_REJOINUNDEF)) {
+            assert(cache->mds->is_rejoin());
+            state_clear(STATE_REJOINUNDEF);
+            cache->opened_undef_dirfrag(this);
+        } else {
+            set_version(1);
+        }
+    }
+} else {
+    set_version(1);
}

11/25/2015 4/5
mark_complete();

if (c)
    cache->mds->queue_waiter(c);
return;

#6 - 05/12/2015 05:07 AM - Greg Farnum
- Assignee set to Zheng Yan

Put it in a PR, please?

#7 - 05/12/2015 08:48 AM - John Spray
- Status changed from New to Pending Backport
- Backport set to hammer

Needs backport to hammer as that's where the issue appeared.

#8 - 05/15/2015 03:53 AM - Zheng Yan
this bug happens only in multimds case. do we need to backport multimds fixes?

#9 - 05/15/2015 03:56 AM - Greg Farnum
Generally speaking no, but unless it's difficult to backport we might as well do so for things people have actually hit outside the lab.

#10 - 05/31/2015 10:59 PM - Loic Dachary
- Project changed from Ceph to fs
- Category deleted (MDS)
- Target version deleted (v0.94)
- Affected Versions deleted (v0.94)

#11 - 06/16/2015 01:49 PM - Zheng Yan
- Status changed from Pending Backport to Resolved

Files
mds.virt-master.log.bz2 319 KB 05/06/2015 Andrey Matyashov