# CephFS - Bug #24137

## client: segfault in trim_caps

05/15/2018 07:18 PM - Patrick Donnelly

| | | | |
|---|---|---|---|
| **Status:** | Resolved | **% Done:** | 0% |
| **Priority:** | Urgent | | |
| **Assignee:** | Patrick Donnelly | | |
| **Category:** | Correctness/Safety | | |
| **Target version:** | v14.0.0 | | |
| **Source:** | Support | **Affected Versions:** | v12.2.4 |
| **Tags:** | | **ceph-qa-suite:** | |
| **Backport:** | mimic,luminous | **Component(FS):** | Client |
| **Regression:** | No | **Labels (FS):** | crash |
| **Severity:** | 2 - major | **Pull request ID:** | |
| **Reviewed:** | | **Crash signature:** | |

**Description**

```
ceph version 12.2.4-XXX (78f60b924802e34d44f7078029a40dbe6c0c922f) luminous (stable)
1: (()+0x6d4d94) [0x560100b8cd94]
2: (()+0x11390) [0x7f530ac3e390]
3: (Inode::get()+0x3c) [0x5601007792ec]
4: (Client::trim_caps(MetaSession*, int)+0x123) [0x56010071b493]
5: (Client::handle_client_session(MClientSession*)+0x331) [0x56010071c581]
6: (Client::ms_dispatch(Message*)+0x4cf) [0x56010074abdf]
7: (DispatchQueue::entry()+0xf4a) [0x560100b2347a]
8: (DispatchQueue::DispatchThread::entry()+0xd) [0x56010085913d]
9: (()+0x76ba) [0x7f530ac346ba]
10: (clone()+0x6d) [0x7f5309a9c41d]
NOTE: a copy of the executable, or `objdump -rdS <executable>` is needed to interpret this.
```

BZ: https://bugzilla.redhat.com/show_bug.cgi?id=1578275

**Related issues:**

| | |
|---|---|
| Copied to CephFS - Backport #24185: luminous: client: segfault in trim_caps | **Resolved** |
| Copied to CephFS - Backport #24186: mimic: client: segfault in trim_caps | **Resolved** |

### History

#### #1 - 05/15/2018 07:38 PM - Patrick Donnelly

Reasonable assumption about this crash is either the inode was deleted (in which case the Cap should have been deleted too) or the inode is a nullptr. The latter should never happen AFAICT as the inode is set only when the cap is created.

I couldn't find a way for the inode/cap to be deleted while the cap remains in the session cap list. Ideas?

#### #2 - 05/16/2018 11:10 AM - Zheng Yan

```
commit 4dda1b6ead6b1f04f996403881f61e9b7d94dba0
Author: Patrick Donnelly <pdonnell@redhat.com>
Date:   Sun Nov 19 15:08:18 2017 -0800

    client: anchor Inode while trimming caps

    This prevents the Inode from being deleted until after cap trimming is
```

```
    finished. In particular, this prevents remove_all_caps from being called which
    screws up the traversal of caps in trim_caps.

    Fixes: http://tracker.ceph.com/issues/22157

    Signed-off-by: Patrick Donnelly <pdonnell@redhat.com>
    (cherry picked from commit 1439337e949c9fcb7d15eb38c22d19eb57d3d0f2)
```

I think above commit isn't quite right. how about patch below

```
diff --git a/src/client/Client.cc b/src/client/Client.cc
index 5bbe449974..2885a87e48 100644
--- a/src/client/Client.cc
+++ b/src/client/Client.cc
@@ -4113,14 +4119,30 @@ void Client::trim_caps(MetaSession *s, uint64_t max)

   uint64_t trimmed = 0;
   auto p = s->caps.begin();
-  std::set<InodeRef> anchor; /* prevent put_inode from deleting all caps during traversal */
-  while ((caps_size - trimmed) > max && !p.end()) {
-    Cap *cap = *p;
-    InodeRef in(cap->inode);
+
+  InodeRef next_in;
+  Cap *next_cap;
+  if (p.end()) {
+    next_cap = nullptr;
+  } else {
+    next_cap = *p;
+    next_in = next_cap->inode;
+  }
+
+  while (next_cap && (caps_size - trimmed) > max) {
+    Cap *cap = next_cap;
+    InodeRef in;
+    in.swap(next_in);

     // Increment p early because it will be invalidated if cap
     // is deleted inside remove_cap
     ++p;
+    if (p.end()) {
+      next_cap = nullptr;
+    } else {
+      next_cap = *p;
+      next_in = next_cap->inode;
+    }

     if (in->caps.size() > 1 && cap != in->auth_cap) {
       int mine = cap->issued | cap->implemented;
@@ -4129,7 +4151,6 @@ void Client::trim_caps(MetaSession *s, uint64_t max)
       if (!(get_caps_used(in.get()) & ~oissued & mine)) {
     ldout(cct, 20) << " removing unused, unneeded non-auth cap on " << *in << dendl;
     cap = (remove_cap(cap, true), nullptr);
-        /* N.B. no need to push onto anchor, as we are only removing one cap */
     trimmed++;
       }
     } else {
@@ -4148,8 +4169,6 @@ void Client::trim_caps(MetaSession *s, uint64_t max)
         // the end of this function.
         _schedule_invalidate_dentry_callback(dn, true);
       }
-          ldout(cct, 20) << " anchoring inode: " << in->ino << dendl;
-          anchor.insert(in);
       trim_dentry(dn);
         } else {
           ldout(cct, 20) << "  not expirable: " << dn->name << dendl;
@@ -4162,8 +4181,6 @@ void Client::trim_caps(MetaSession *s, uint64_t max)
       }
     }
   }
-  ldout(cct, 20) << " clearing anchored inodes" << dendl;
-  anchor.clear();
```

```
  caps_size = s->caps.size();
  if (caps_size > (size_t)max)
```

#### #3 - 05/16/2018 06:16 PM - Patrick Donnelly

Zheng Yan wrote:

> [...]
>
> I think above commit isn't quite right. how about patch below
>
> [...]

I'm not seeing how that helps the situation. The issue my patch was trying to solve is preventing the next cap (what p points to after ++p) from being invalidated due to trim_dentry -> Client::unlink -> Dentry::put -> Inode::put -> intrusive_ptr_release -> Client::put_inode -> Client::remove_all_caps. This should be prevented by anchoring the Inode references as in the original PR?

We need a test case to definitively show the issue. I'm not willing to push any new fixes for this without a reproducer.

#### #4 - 05/17/2018 12:44 AM - Zheng Yan

The problem is that anchor only pins current inode. Client::unlink() still may drop reference of its parent inode.

#### #5 - 05/17/2018 04:44 AM - Patrick Donnelly

Zheng Yan wrote:

> The problem is that anchor only pins current inode. Client::unlink() still may drop reference of its parent inode.

Okay that makes sense I think. We still need a test that demonstrates the problem.

#### #6 - 05/17/2018 08:44 AM - Zheng Yan

*- File ceph-client.27063.log added*

*- File test_trim_caps.cc added*

compile test_trim_caps.cc with the newest libcephfs. set mds_min_caps_per_client to 1, set mds_max_ratio_caps_per_client to 0. then run test_trim_caps

```
[Current thread is 1 (Thread 0x7fffd97fa700 (LWP 27088))]
Missing separate debuginfos, use: dnf debuginfo-install glibc-2.25-13.fc26.x86_64 libblkid-2.30.2-1.fc26.x86_64 libgcc-7.3.1-2.fc26.x86_64 libibverbs-1.2.1-4.fc26.x86_64 libnl3-3.3.0-1.fc26.x86_64 libstdc++-7.3.1-2.fc26.x86_64 libuuid-2.30.2-1.fc26.x86_64 lttng-ust-2.9.0-2.fc26.x86_64 nspr-4.19.0-1.fc26.x86_64 nss-3.36.1-1.0.fc26.x86_64 nss-softokn-3.36.1-1.0.fc26.x86_64 nss-softokn-freebl-3.36.1-1.0.fc26.x86_64 nss-util-3.36.1-1.0.fc26.x86_64 sqlite-libs-3.20.1-2.fc26.x86_64 userspace-rcu-0.9.3-2.fc26.x86_64 zlib-1.2.11-2.fc26.x86_64
(gdb) bt
#0  0x00007fffd40004d8 in ?? ()
#1  0x00007fffee48bd9b in ceph::logging::log_clock::now (this=0x7fffd40004e8) at /home/zhyan/Ceph/ceph-2/src/log/LogClock.h:95
#2  ceph::logging::Log::create_entry (this=0x7fffd40004b8, level=level@entry=15, subsys=subsys@entry=21, expected_size=expected_size@entry=0x7ffff7dd5ed0 <Inode::get()::_log_exp_length>)
    at /home/zhyan/Ceph/ceph-2/src/log/Log.cc:263
#3  0x00007ffff7b82efd in Inode::get (this=0x7fffd4014820) at /home/zhyan/Ceph/ceph-2/src/client/Inode.cc:383
#4  0x00007ffff7aec2ca in intrusive_ptr_add_ref (in=<optimized out>) at /home/zhyan/Ceph/ceph-2/src/client/Client.cc:13974
#5  0x00007ffff7b2188a in boost::intrusive_ptr<Inode>::intrusive_ptr (add_ref=true, p=<optimized out>, this=0x7fffd97f75d8) at /mnt/misc/Ceph/ceph-2/build/boost/include/boost/smart_ptr/intrusive_ptr.hpp:69
#6  Client::trim_caps (this=this@entry=0x77edd0, s=s@entry=0x7958a8, max=1) at /home/zhyan/Ceph/ceph-2/src/client/Client.cc:4124
#7  0x00007ffff7b22652 in Client::handle_client_session (this=this@entry=0x77edd0, m=m@entry=0x7fffe4000f00) at /home/zhyan/Ceph/ceph-2/src/client/Client.cc:2102
#8  0x00007ffff7b5461f in Client::ms_dispatch (this=0x77edd0, m=0x7fffe4000f00) at /home/zhyan/Ceph/ceph-2/src/client/Client.cc:2544
#9  0x00007fffee4e3b4b in Messenger::ms_deliver_dispatch (m=0x7fffe4000f00, this=0x78c750) at /home/zhyan/Ceph/ceph-2/src/msg/Messenger.h:667
#10 DispatchQueue::entry (this=0x78c8d0) at /home/zhyan/Ceph/ceph-2/src/msg/DispatchQueue.cc:201
#11 0x00007fffee5862dd in DispatchQueue::DispatchThread::entry (this=<optimized out>) at /home/zhyan/Ceph/ceph-2/src/msg/DispatchQueue.h:101
#12 0x00007fffec76436d in start_thread () from /lib64/libpthread.so.0
#13 0x00007ffff6f47b4f in clone () from /lib64/libc.so.6
```

#### #7 - 05/18/2018 12:14 AM - Patrick Donnelly

*- Status changed from New to In Progress*

*- Assignee set to Patrick Donnelly*

https://github.com/ceph/ceph/pull/22073

#### #8 - 05/19/2018 04:37 AM - Patrick Donnelly

*- Target version changed from v13.2.0 to v14.0.0*

*- Backport changed from luminous to mimic,luminous*

#### #9 - 05/19/2018 04:38 AM - Patrick Donnelly

*- Status changed from In Progress to Pending Backport*

#### #10 - 05/19/2018 04:45 AM - Patrick Donnelly

*- Copied to Backport #24185: luminous: client: segfault in trim_caps added*

#### #11 - 05/19/2018 10:04 AM - Nathan Cutler

*- Copied to Backport #24186: mimic: client: segfault in trim_caps added*

#### #12 - 06/20/2018 10:01 PM - Nathan Cutler

*- Status changed from Pending Backport to Resolved*

### Files

| | | | |
|---|---|---|---|
| ceph-client.27063.log | 87.7 KB | 05/17/2018 | Zheng Yan |
| test_trim_caps.cc | 1.59 KB | 05/17/2018 | Zheng Yan |