CephFS - Bug #19635

Deadlock on two ceph-fuse clients accessing the same file

04/16/2017 09:48 PM - John Spray

Status:	Resolved	**% Done:**	0%
Priority:	Normal		
Assignee:	Zheng Yan		
Category:	Correctness/Safety		
Target version:	v12.1.0		
Source:		**Affected Versions:**	
Tags:		**ceph-qa-suite:**	
Backport:	jewel, kraken	**Component(FS):**	
Regression:	No	**Labels (FS):**	
Severity:	3 - minor	**Pull request ID:**	
Reviewed:		**Crash signature:**	

Description

See Dan's reproducer script, and thread "[ceph-users] fsping, why you no work no mo?"
https://raw.githubusercontent.com/dvanders/fsping/

When I started a vstart cluster and mounted two fuse clients, then ran the script, I got two blocked requests like this

```
(virtualenv) jspray@senta04:~/ceph/build$ bin/ceph daemon mds.a ops
*** DEVELOPER MODE: setting PATH, PYTHONPATH and LD_LIBRARY_PATH ***
{
    "ops": [
        {
            "description": "client_request(client.4110:27 lookup #1/senta04.ack 2017-04-16 17:39:0
9.476736 caller_uid=1121, caller_gid=1121{})",
            "initiated_at": "2017-04-16 17:39:09.476974",
            "age": 486.457417,
            "duration": 486.457469,
            "type_data": [
                "failed to rdlock, waiting",
                "client.4110:27",
                "client_request",
                {
                    "client": "client.4110",
                    "tid": 27
                },
                [
                    {
                        "time": "2017-04-16 17:39:09.476974",
                        "event": "initiated"
                    },
                    {
                        "time": "2017-04-16 17:39:09.486978",
                        "event": "failed to rdlock, waiting"
                    }
                ]
            ]
        },
        {
            "description": "client_request(client.4111:10 getattr pAsLsXsFs #100000003e9 2017-04-1
6 17:39:09.488176 caller_uid=1121, caller_gid=1121{})",
            "initiated_at": "2017-04-16 17:39:09.488318",
            "age": 486.446072,
            "duration": 486.446188,
```

```
            "type_data": [
                "failed to rdlock, waiting",
                "client.4111:10",
                "client_request",
                {
                    "client": "client.4111",
                    "tid": 10
                },
                [
                    {
                        "time": "2017-04-16 17:39:09.488318",
                        "event": "initiated"
                    },
                    {
                        "time": "2017-04-16 17:39:09.489099",
                        "event": "failed to rdlock, waiting"
                    }
                ]
            ]
        }
    ],
    "num_ops": 2
}
```


This is apparently something that worked in 10.2.5 and is now failing on more recent versions.

Related issues:

Copied to CephFS - Backport #20027: jewel: Deadlock on two ceph-fuse clients ...	**Resolved**
Copied to CephFS - Backport #20028: kraken: Deadlock on two ceph-fuse clients...	**Resolved**

History

#1 - 04/16/2017 11:02 PM - John Spray

I was wondering if d463107473 ("mds: finish lock waiters in the same order that they were added.") could have been the cause, but the issue still happens if I revert that.

#2 - 04/17/2017 12:15 AM - John Spray

Those requests are getting hung up on the iauth and ixattr locks on the inode for the ".syn" file the test script creates -- those locks are in the excl->sync transition at the time.

When we go into that transition we're not sending any revokes to the clients, but the server seems to be acting kind of like it's sat there waiting for a caps message maybe. Hmm.

#3 - 04/17/2017 10:18 AM - Zheng Yan

- Assignee set to Zheng Yan

#4 - 04/17/2017 11:13 AM - Zheng Yan

- Status changed from New to In Progress

This bug happens in following sequence of events

- Reuqest1 (from client1) create file1 (mds issues caps Asx to client1, early reply is no allowed)
- Request2 (from client2) lookup file1 (dentry lock is xlocked, waiting)
- Log event of request1 get journaled (Server::reply_client_request() calls MDCache::request_drop_non_rdlocks(). Request2 get dispatched when dropping xlock of the dentry lock)
- Request2 reovkes caps Ax from client1. (the caps haven't beeb sent to client. so Lock::issue_caps() just updates client1's caps. the caps get updated, but Locker::eval_gather() is not called. request2 waits infinitely)

- Send reply of request1 to client1 (with the update caps)

I think we should avoid finishing contexts directly when drop locks. (queue contexts to finisher instead)

#5 - 04/24/2017 03:01 AM - Zheng Yan

- Status changed from In Progress to Fix Under Review

https://github.com/ceph/ceph/pull/14743

#6 - 05/11/2017 09:46 AM - John Spray

- Status changed from Fix Under Review to Pending Backport

- Backport set to jewel, kraken

#7 - 05/22/2017 10:29 AM - Nathan Cutler

- Copied to Backport #20027: jewel: Deadlock on two ceph-fuse clients accessing the same file added

#8 - 05/22/2017 10:29 AM - Nathan Cutler

- Copied to Backport #20028: kraken: Deadlock on two ceph-fuse clients accessing the same file added

#9 - 07/19/2017 01:43 PM - Nathan Cutler

- Status changed from Pending Backport to Resolved