fs - Bug #23291

client: add way to sync setattr operations to MDS

03/09/2018 04:19 PM - Jeff Layton

Status:	Resolved	**% Done:**	0%
Priority:	Normal		
Assignee:	Jeff Layton		
Category:	Correctness/Safety		
Target version:	v13.0.0		
Source:	Development	**Affected Versions:**	
Tags:		**ceph-qa-suite:**	
Backport:	luminous	**Component(FS):**	Client, libcephfs
Regression:	No	**Labels (FS):**	
Severity:	3 - minor	**Pull request ID:**	
Reviewed:		**Crash signature:**	

Description

Zheng pointed out that we could end up with a setattr request being cached in the libcephfs client after it has replied with success to the application. This is problematic for ganesha:

1/ NFS client does a NFS SETATTR
2/ ganesha issues setattrx to libcephfs, result is cached in client and not sent to MDS
3/ ganesha replies with success to client
4/ ganesha crashes

setattrx changes are then lost.

NFS expects setattr requests to be durable. The right fix is probably to allow the application to request that a setattrx be forced through to the MDS. In practice, ganesha and samba will almost never want the client to cache the results of a setattr.

In the libcephfs code the decision to cache the change is done in _do_setattr. We just need to provide some way to force the change to go to the MDS, even when it holds the appropriate caps. Alternately, we could just always have ceph_ll_setattr do that. AFAIK, ganesha is the only caller of that function so far.

Related issues:

Copied to fs - Backport #23474: luminous: client: allow caller to request tha...	**Resolved**

History

#1 - 03/09/2018 04:20 PM - Jeff Layton

- Description updated

#2 - 03/09/2018 10:23 PM - Patrick Donnelly

Jeff Layton wrote:

> Zheng pointed out that we could end up with a setattr request being cached in the libcephfs client after it has replied with success to the application. This is problematic for ganesha:
>
> 1/ NFS client does a NFS SETATTR
> 2/ ganesha issues setattrx to libcephfs, result is cached in client and not sent to MDS
> 3/ ganesha replies with success to client
> 4/ ganesha crashes
>
> setattrx changes are then lost.
>
> NFS expects setattr requests to be durable.

That requires synchronous requirements. e.g. fchmod(fd, buf); fsync(fd);

Is that what Ganesha does for the local file system?

> The right fix is probably to allow the application to request that a setattrx be forced through to the MDS. In practice, ganesha and samba will almost never want the client to cache the results of a setattr.

To satisfy durability requirements, the client must also wait for the safe (i.e. durable) reply. Right?

#3 - 03/09/2018 10:24 PM - Patrick Donnelly

- Subject changed from Allow caller to request that setattr request be synchronous to client: allow caller to request that setattr request be synchronous

- Category set to Correctness/Safety

- Target version set to v13.0.0

- Source set to Development

- Component(FS) Client added

#4 - 03/09/2018 11:34 PM - Patrick Donnelly

- Related to Bug #16739: Client::setxattr always sends setxattr request to MDS added

#5 - 03/09/2018 11:34 PM - Patrick Donnelly

Jeff, see also #16739. Maybe that's why it hasn't been a problem?

#6 - 03/12/2018 02:11 PM - Jeff Layton

No. I don't think that bug is related. That one is dealing with setxattr (setting extended attributes) and yes, the client does always send those to the MDS. This is more about changing inode metadata (truncate, chown, chmod, etc.) and we do cache those changes locally if we have the right caps.

#7 - 03/12/2018 02:15 PM - Jeff Layton

- Related to deleted (Bug #16739: Client::setxattr always sends setxattr request to MDS)

#8 - 03/12/2018 02:34 PM - Jeff Layton

The big question here is how to achieve this:

The kernel has an AT_STATX_FORCE_SYNC flag, and we could repurpose it for use in setattr commands. Ganesha however uses ceph_ll_setattr, and that does not accept a flags field. We'd have to add a variant that does, which would be an API change (at a minimum), and then handle that at build-time in ganesha.

Alternately, we could have some conf setting in the client that forces setattrs to be synchronous. In practice, we will always want these to be synchronous from Ganesha anyway so a client-wide setting would be fine for this purpose.

#9 - 03/12/2018 07:53 PM - Patrick Donnelly

Jeff Layton wrote:

> Zheng pointed out that we could end up with a setattr request being cached in the libcephfs client after it has replied with success to the application. This is problematic for ganesha:

> 1/ NFS client does a NFS SETATTR
> 2/ ganesha issues setattrx to libcephfs, result is cached in client and not sent to MDS
> 3/ ganesha replies with success to client
> 4/ ganesha crashes
>
> setattrx changes are then lost.
>
> NFS expects setattr requests to be durable. The right fix is probably to allow the application to request that a setattrx be forced through to the MDS. In practice, ganesha and samba will almost never want the client to cache the results of a setattr.

Hmm, so NFS requires that setattr requests against the backing FS be as if:

```
   setattr(fd, ...);
   fsync(fd);
```

? Why not just use ceph_fsetattrx followed by ceph_fsync? Do we really need another API?

#10 - 03/12/2018 08:20 PM - Jeff Layton

ceph_ll_fsync would give us almost the semantics we need. The main problem is that we may not have the file open and all of the variants currently take a Fh *. We need one that takes an Inode *.

Maybe we should add:

```
ceph_ll_sync_inode(Inode *i, bool syncdataonly);
```

?

#11 - 03/12/2018 09:07 PM - Patrick Donnelly

Sounds good to me.

#12 - 03/14/2018 07:11 PM - Jeff Layton

Ok, I have a draft patch for this now. What I don't have is a great way to test it.

Hmm...now that I look, we have no test coverage at all for functional ceph_fsync or ceph_ll_fsync. I'll have to give that some thought.

#13 - 03/20/2018 11:35 AM - Jeff Layton

- Status changed from New to In Progress

PR is here:

https://github.com/ceph/ceph/pull/20913

#14 - 03/27/2018 09:58 PM - Patrick Donnelly

- Status changed from In Progress to Pending Backport

- Backport set to luminous

#15 - 03/28/2018 05:51 AM - Nathan Cutler

- Copied to Backport #23474: luminous: client: allow caller to request that setattr request be synchronous added

#16 - 04/06/2018 02:00 PM - Jeff Layton

- Subject changed from client: allow caller to request that setattr request be synchronous to client: add way to sync setattr operations to MDS

#17 - 05/09/2018 06:57 PM - Nathan Cutler

- Status changed from Pending Backport to Resolved