

## CephFS - Bug #41327

### mds: dirty rstat lost during scatter-gather process

08/18/2019 01:55 PM - Xuehan Xu

<b>Status:</b>	Fix Under Review	<b>% Done:</b>	0%
<b>Priority:</b>	Normal		
<b>Assignee:</b>	Xuehan Xu		
<b>Category:</b>	Correctness/Safety		
<b>Target version:</b>	v17.0.0		
<b>Source:</b>	Community (dev)	<b>ceph-qa-suite:</b>	
<b>Tags:</b>		<b>Component(FS):</b>	
<b>Backport:</b>	pacific,octopus,nautilus	<b>Labels (FS):</b>	
<b>Regression:</b>	No	<b>Pull request ID:</b>	29721
<b>Severity:</b>	3 - minor	<b>Crash signature (v1):</b>	
<b>Reviewed:</b>		<b>Crash signature (v2):</b>	
<b>Affected Versions:</b>			

#### Description

In the following scenario, the current lock's dirty state could be lost:

1. current lock's state is LOCK\_LOCK;
2. 2. auth send LOCK\_AC\_MIX, and the current lock do finish\_scatter\_update, which wrlock\_force the current lock;
3. 3. auth send LOCK\_AC\_LOCK, and would be forced to wait, since the current lock is wrlocked;
4. 4. the child inode want to pop rstat up to the current lock's parent, which would also be forced to wait since the current lock state is LOCK\_MIX\_LOCK; the current lock would be marked dirty, and the child inode would be attached to the corresponding dir's dirty\_rstat\_inodes;
5. 5. the previous finish\_scatter\_update finishes, the codes reaches here and "start\_flush" the current lock, which would clear the dirty flag of the current lock.

As the CInode::encode\_lock\_state method doesn't pop dirty\_rstat\_inodes' rstat to the corresponding dir, step 5 actually lead to the lost of the dirty state marked by step 4.

#### History

##### #1 - 08/19/2019 01:40 PM - Patrick Donnelly

- Status changed from New to Fix Under Review

- Assignee set to Xuehan Xu

##### #2 - 02/26/2020 03:21 PM - Patrick Donnelly

- Target version changed from v15.0.0 to v16.0.0

##### #3 - 01/15/2021 10:48 PM - Patrick Donnelly

- Target version changed from v16.0.0 to v17.0.0

- Backport set to pacific,octopus,nautilus