

CephFS - Bug #39987

mds: MDCache::cow_inode does not cleanup unneeded client_snap_caps

05/21/2019 09:58 AM - Zheng Yan

Status: Resolved	% Done: 0%
Priority: High	
Assignee: Zheng Yan	
Category:	
Target version: v15.0.0	
Source: Community (user)	ceph-qa-suite:
Tags:	Component(FS): MDS
Backport: nautilus,mimic	Labels (FS):
Regression: No	Pull request ID: 28190
Severity: 3 - minor	Crash signature (v1):
Reviewed:	Crash signature (v2):
Affected Versions:	
Description	
An user reported a bug that mds couldn't finish freezing dirfrag. Cache dump includes following entries.	
<pre>[inode 0x1000056a9ad [112,head] /hpc/home/neda/ffpy_test/ffpy/interp_test/smooth_z0/z0meso.nc auth v5478173 s=2319190 n(v0 rc2019-05-18 07:15:27.742396 b2319190 l=1+0) (iversion lock) caps={62472=pAsLsXsFscr/-@115} ptrwaiter=0 request=0 lock=0 caps=1 truncating=0 needsnapflush=0 dirtyparent=0 dirty=0 waiter=0 authpin=0 0x55c2475ed800] ... [inode 0x1000056a9ad [10c,10d] /hpc/home/neda/ffpy_test/ffpy/interp_test/smooth_z0/z0meso.nc auth v5478065 s=2319190 n(v0 rc2019-05-17 19:36:50.430298 b2319190 l=1+0) (iversion lock) 0x55c24feeae0 0] [inode 0x1000056a9ad [10e,111] /hpc/home/neda/ffpy_test/ffpy/interp_test/smooth_z0/z0meso.nc auth v5478171 ap=3+0 s=2319190 n(v0 rc2019-05-18 07:15:27.742396 b2319190 l=1+0)/n(v0 rc2019-05-17 19:36:50.430298 b2319190 l=1+0) (iauth snap->sync w=1) (ifile snap->sync w=1) (ixattr snap->sync w=1) (iversion lock) ptrwaiter=0 request=0 lock=3 dirty=0 authpin=1 0x55c24c07d800]</pre>	
It's likely that inode 0x1000056a9ad [10c,111] was COWed and head_in->split_need_snapflush contained no items in range [10e, 111]	
Related issues:	
Duplicated by CephFS - Bug #42338: file system keeps on deadlocking with unre...	Duplicate 10/16/2019
Copied to CephFS - Backport #40444: mimic: mds: MDCache::cow_inode does not c...	Resolved
Copied to CephFS - Backport #40445: nautilus: mds: MDCache::cow_inode does no...	Resolved

History

#1 - 05/29/2019 01:47 PM - Patrick Donnelly

- Priority changed from Normal to High
- Target version set to v15.0.0
- Start date deleted (05/21/2019)
- Source set to Community (user)
- Component(FS) MDS added

#2 - 05/29/2019 01:53 PM - Zheng Yan

- Status changed from New to Fix Under Review

- Pull request ID set to 28190

#3 - 06/08/2019 04:19 AM - Patrick Donnelly

- Status changed from Fix Under Review to Pending Backport

- Assignee set to Zheng Yan

#4 - 06/09/2019 03:24 AM - Zheng Yan

- Status changed from Pending Backport to Fix Under Review

#5 - 06/09/2019 03:51 AM - Zheng Yan

<https://github.com/ceph/ceph/pull/28190> is incomplete

<https://github.com/ceph/ceph/pull/28459>

#6 - 06/19/2019 01:35 AM - Patrick Donnelly

- Status changed from Fix Under Review to Pending Backport

#7 - 06/19/2019 06:04 PM - Nathan Cutler

- Copied to Backport #40444: *mimic: mds: MDCache::cow_inode does not cleanup unneeded client_snap_caps added*

#8 - 06/19/2019 06:04 PM - Nathan Cutler

- Copied to Backport #40445: *nautilus: mds: MDCache::cow_inode does not cleanup unneeded client_snap_caps added*

#9 - 09/23/2019 05:16 PM - Nathan Cutler

When there are multiple PRs fixing a single tracker, it's a good idea to "unset" (depopulate) the Pull request ID field. When that field is populated, it's easy for the backporter to miss the follow-on PR (PRs).

#10 - 09/24/2019 07:03 AM - Patrick Donnelly

Nathan Cutler wrote:

When there are multiple PRs fixing a single tracker, it's a good idea to "unset" (depopulate) the Pull request ID field. When that field is populated, it's easy for the backporter to miss the follow-on PR (PRs).

FWIW, I don't blame you for this Nathan. I plan to go a step further and not permit a tracker ticket to go backwards like this, i.e. from PB back to NR. Instead, we'll create a new tracker ticket and note the issue in the broken backport ticket.

#11 - 09/25/2019 09:29 AM - Nathan Cutler

I plan to go a step further and not permit a tracker ticket to go backwards like this, i.e. from PB back to NR. Instead, we'll create a new tracker ticket and note the issue in the broken backport ticket.

@Patrick Sometimes it's necessary to create backport issues in advance (before the issue enters PB status), and I had been doing this by setting PB

temporarily. With the new, stricter workflow you just described (which I fully support BTW) it will no longer be possible to set PB "temporarily", so I added a --force option to backport-create-issue which will make it create backport issues regardless of the issue status. See <https://github.com/ceph/ceph/pull/30571> for details.

#12 - 09/25/2019 09:36 AM - Nathan Cutler

Oh, and one more thing: issues in Resolved status can be reverted to Need Review (or In Progress, or even New) as well.

#13 - 09/26/2019 10:05 AM - Patrick Donnelly

Nathan Cutler wrote:

I plan to go a step further and not permit a tracker ticket to go backwards like this, i.e. from PB back to NR. Instead, we'll create a new tracker ticket and note the issue in the broken backport ticket.

@Patrick Sometimes it's necessary to create backport issues in advance (before the issue enters PB status), and I had been doing this by setting PB temporarily. With the new, stricter workflow you just described (which I fully support BTW) it will no longer be possible to set PB "temporarily", so I added a --force option to backport-create-issue which will make it create backport issues regardless of the issue status. See <https://github.com/ceph/ceph/pull/30571> for details.

I wasn't planning a technical change to redmine to enforce this policy but if you find a way to do it I'd support it.

#14 - 10/11/2019 10:07 AM - Nathan Cutler

- Status changed from Pending Backport to Resolved

While running with --resolve-parent, the script "backport-create-issue" noticed that all backports of this issue are in status "Resolved" or "Rejected".

#15 - 10/25/2019 12:16 AM - Patrick Donnelly

- Duplicated by Bug #42338: file system keeps on deadlocking with unresolved slow requests (failed to authpin, subtree is being exported) added