

## CephFS - Feature #48991

### client: allow looking up snapped inodes by inode number+snapid tuple

01/25/2021 08:10 PM - Jeff Layton

<b>Status:</b>	Pending Backport	<b>% Done:</b>	0%
<b>Priority:</b>	Normal		
<b>Assignee:</b>	Jeff Layton		
<b>Category:</b>			
<b>Target version:</b>	v17.0.0		
<b>Source:</b>	Development	<b>Affected Versions:</b>	
<b>Tags:</b>		<b>Component(FS):</b>	Client, Ganesha FSAL, libcephfs
<b>Backport:</b>	pacific,octopus,nautilus	<b>Labels (FS):</b>	
<b>Reviewed:</b>		<b>Pull request ID:</b>	39294
<b>Description</b>			
Currently, we have <code>ceph_ll_lookup_inode()</code> , but that only takes an inode number and can't deal with a snapped inode. Add a new lookup function to <code>libcephfs</code> that allows you to look up snapped or non-snapped inodes by inode number and snapid.			
This is necessary to allow <code>ganesha</code> to work properly when dealing with snapped inodes.			
<b>Related issues:</b>			
Copied to CephFS - Backport #49512: pacific: client: allow looking up snapped...		<b>Resolved</b>	
Copied to CephFS - Backport #49513: octopus: client: allow looking up snapped...		<b>In Progress</b>	
Copied to CephFS - Backport #49514: nautilus: client: allow looking up snapped...		<b>Resolved</b>	

## History

### #1 - 02/01/2021 02:41 PM - Douglas Fuller

- Assignee set to Jeff Layton

### #2 - 02/01/2021 06:11 PM - Jeff Layton

I started looking at this today, and it's a little trickier than I thought. The current client code that sends the `LOOKUPINO` request (`_lookup_ino`) always just leaves the `snapid` field zeroed out (AFAICT).

The corresponding handler on the MDS (`Server::handle_client_lookup_ino`), seems to rely on that though:

```
if ((uint64_t)req->head.args.lookupino.snapid > 0)
    return _lookup_snap_ino(mdr);
```

That condition is weird, in that "snapid" is `le64`, so this is effectively casting that to `u64` (regardless of host endianness), and the if statement just a check to see whether that value `!= 0`. `CEPH_NOSNAP`, however is defined as `(__u64)-2`, which will evaluate as being `>0` in the above conditional.

It's not clear why this is testing for 0 instead of `CEPH_NOSNAP`. I guess I need to understand why `CEPH_NOSNAP` is -2 and not 0. What is the special significance of `snapid` being 0 in the if condition in `Server::handle_client_lookup_ino`?

**#3 - 02/17/2021 09:19 PM - Patrick Donnelly**

- Status changed from New to Fix Under Review
- Target version set to v17.0.0
- Source set to Development
- Backport set to *pacific,octopus,nautilus*
- Pull request ID set to 39294

**#4 - 02/26/2021 05:29 PM - Patrick Donnelly**

- Tracker changed from Bug to Feature
- Subject changed from *allow looking up snapped inodes by inode number+snapid tuple* to *client: allow looking up snapped inodes by inode number+snapid tuple*
- Status changed from Fix Under Review to Pending Backport
- Component(FS) Client added

**#5 - 02/26/2021 05:30 PM - Backport Bot**

- Copied to Backport #49512: *pacific: client: allow looking up snapped inodes by inode number+snapid tuple* added

**#6 - 02/26/2021 05:30 PM - Backport Bot**

- Copied to Backport #49513: *octopus: client: allow looking up snapped inodes by inode number+snapid tuple* added

**#7 - 02/26/2021 05:30 PM - Backport Bot**

- Copied to Backport #49514: *nautilus: client: allow looking up snapped inodes by inode number+snapid tuple* added