NetApp

Go further, faster®

RFC3530bis Home Stretch

Thomas Haynes

NetApp

Home stretch

- Task List
  - 54 items
    - 16 items remain
    - +2 to be discussed later
  - https://github.com/loghyr/3530bis/blob/master/tasklist.txt
- Do we need a review of draft after all edits?

Walk through the task list

- Is it an issue we want to fix?
  - Can we describe the issue?
  - How do different implementations resolve it?

Item #4

- What are "courtesy locks"?
- What are implementations doing?
- What do we want to do?

NetApp

Item #9

- 9) We should clarify the passage on NFS4ERR_LOCKS_HELD. Since lock owners and open owners are different things, we need to clarify which locks may trigger an NFS4ERR_LOCKS_HELD error on a CLOSE for a given open_stateid. In the same vein, which locks does section 9.8 allow the server to free when the client does a CLOSE?
- This all harks back to the mailing list discussions about lock_stateids being <per-file,per-open_owner,per-lock_owner> vs. just being <per-file,per-lock_owner>
- Add DOWNGRADE here?? Dave

Item #2 (subset of 9?)

- Explain what associated means in this context:
- In sec. 14.2.2 DESCRIPTION
- The share reservations and other state information released at the server as a result of this CLOSE is only associated with the supplied stateid.

Item #10

- Recommendations/workarounds for dealing with the OPEN and CB_RECALL race.
  - How are different implementations handling it?

Item #43

- How to interpret empty path components in fs_locations/fs_locations_info

Item #11

- Clarify the use of the delegation vs lock vs open stateid in SETATTR.
- Item #12
  - Clarify the various scenarios that should result in a delegation recall (mainly SETATTR by self).

Item #17

- Description of Third Edge Condition

The third edge condition needs to be described. Basically, this is the situation in which a client thinks he has locks because he rebooted in the middle of reclaim without reclaiming those locks. If the client comes back and those locks were taken by another client and then released before that final reboot, the client could reclaim the locks unaware that it had lost them in the interim and could not be guaranteed to continuous holding of the locks.

Item #20

- Explanatory Text on Stateid Contradicts Self and Rest of Protocol

Item #27

- Setattr for a Write Delegated File Should be Done with theDelegation Stateid

- If a client holding a write delegation does a setattr for that file for something other than set size and chooses to use a special stateid to do the setattr, the delegation can be recalled by the server. This is because there is no way for the server to know that the setattr is coming from the client which is holding the delegation, unless it looks at the clientid to figure this out. It would be better if the client sends any such setattrs with the delegation stateid so that the server can avoid recalling the delegation for such requests.

- Pretty straightforward - accept as is?

Item #28

- Reclaiming Local Opens with CLAIM_DELEGATE_CUR Instead of CLAIM_PREVIOUS

If a client holds some local open/lock state while holding a delegation for that file and the server reboots, the client can reclaim this state by sending open requests with the claim_type set to CLAIM_PREVIOUS. However there is no way to differentiate between the reclaim for a regular open and that of a DELEGATE_CUR open. The server can figure out that the open being reclaimed is a DELEGATE_CUR open by comparing the clientid for that open with the client holding the delegation in order to avoid recalling the delegation but it would be better if the client either does not reclaim this open (since it already has the delegation) or reclaim this by setting the claim_type to CLAIM_DELEGATE_CUR so that the server can allow this open to succeed just like a normal CLAIM_DELEGATE_CUR open.

Item #35

- OPEN_DOWNGRADE and posix byte range

Item #13

- Perhaps clarify how clients should be managing delegations? There appear to be server implementations that expect clients to limit the number of delegations they are holding.
- Notes: Clients need to implement mechanism to limit number of held delegations.