



Lost Requests and Replay Logic

Eric Kustarz
Sun Microsystems

Section 8.7

- Lock request may time out or user might abort it
- Server may have received request
- Client's state now out of sync with server
-
- To sync up:
 - Reissue the “lost” request
 - “Undo” the request
-
- Requirement:
 - No dangling state on the server



What is a replay request?

- Is it merely a <seqid, op, owner/stateid> triplet?
 - Is this begging to be spoofed?
- Or entire request?
 - What about OPEN w/ create?



What operations does this affect?

- OPEN
- OPEN_DOWNGRADE
- CLOSE
- LOCK
- LOCKU
- OPEN_CONFIRM?



Solaris Recovery Framework

- Two ways to handle “lost” requests:
 - Immediate
 - Passive
- Upon detection of “lost” request:
 - Kick off recovery
 - Block non-recovery requests
- NFS layer



Solaris Implementation

- OPEN:
 - Reissue and single undo
- OPEN_DOWNGRADE/CLOSE
 - Reissue
- LOCKU
 - Reissue
- LOCK
 - Reissue and (multiple) undoes



Recovery of “lost” lock Example 1

- LOCK(R 0:10, seqid=1)----->
□ <----- OK
- LOCK(R 20:10, seqid=2)----->
□ <----- OK
- LOCK(W 5:20, seqid=3)----->
□ X ---- OK
- Resend LOCK(W 5:20, seqid=3)----->
□ <----- OK
- Reinstate LOCK(R 5:5, seqid=4)----->
□ <----- OK
- Reinstate LOCK(R 20:5, seqid=5)----->
□ <----- OK



Recovery of “lost” lock Example 2

- LOCK(R 0:10, seqid=1)----->
□ <----- OK
- LOCK(W 5:10, seqid=2)----->
□ X ----- OK
- Resend LOCK(W 5:10, seqid=2)----->
□ <----- OK
- Reinstate LOCK(R 5:5, seqid=3)----->
□ <----- OK
- Reinstate LOCKU(10:5, seqid=4) ----->
□ <----- OK



Further Complications

- How much can effort are you willing to do?
 - Recovery errors (STALE_STATEID) vs. recovery errors that you can't really recover from (BAD_SEQID) vs. non-recovery errors (ACCESS)
- Can send SIGLOST but there's no SIG_OPEN_LOST
-
- Does this count as a retry as specified in section 3.1.1?



Differences from v3

- No OPEN/CLOSE state
-
- NFSv3/NLM does not do under the covers “fixing up”
 - Mark the file as having a possible “orphan”
 - Issue a LOCKU for the entire file on CLOSE



Possible Future Enhancements

- NFS4ERR_BAD_REPLAY_REQUEST
-
-
- Bit/flag in state modifying operations stating:
 - “ This is a replay request, if you got this request before, return what you sent back for the original; otherwise, return NFS4ERR_NO_REPLAY”

