NFS Version 4 WG
59th IETF

Tom Talpey - all around good fellow

Brian Pawlowski – co-chair

Spencer Shepler – co-chair

Things

- Sign the “blue sheets” – name and e-mail address

Note Well

All statements related to the activities of the IETF and addressed to the IETF are subject to all provisions of Section 10 of RFC 2026, which grants to the IETF and its participants certain licenses and rights in such statements. Such statements include verbal statements in IETF meetings, as well as written and electronic communications made at any time or place, which are addressed to:

- the IETF plenary session,
- any IETF working group or portion thereof,
- the IESG or any member thereof on behalf of the IESG,
- the IAB or any member thereof on behalf of the IAB,
- any IETF mailing list, including the IETF list itself, any working group or design team list, or any other list functioning under IETF auspices,
- the RFC Editor or the Internet-Drafts function

Statements made outside of an IETF meeting, mailing list or other function, that are clearly not intended to be input to an IETF activity, group or function, are not subject to these provisions.

Agenda

- Welcome and Introduction (Talpey) 1 min
- Agenda bash
  - Blue Sheets
  - NOTE WELL
- Connectathon results (Shepler by proxy) 15 min

  NFS V4 interoperability testing
- Review and discussion of (Talpey) 30 min

  NFS/RDMA Problem Statement, Requirements
- Potential Minor Version Work (Talpey) 20 min

  Review existing items

  Informational items:

  Parallel NFS (pNFS) position statement
- Open discussion/Wrap-up (Talpey)

NFS Version 4 Testing

Connectathon 2004

- Seven NFSv4 testers
  - Sun (Solaris client, server)
  - NetApp (server)
  - Hummingbird (Windows client, server)
  - IBM (AIX client, server)
  - CITI (Linux client, server)
  - EMC (server)
  - HP (server)
- Most ever

Summary

- NFS Version 4 testing went smoothly
- No protocol issues or spec interpretation issues found
- All implementations were testing Kerberos
- Some testing of "advanced" features
  - ACLs
  - Delegations
- Interest in continuing Bake-a-thons

Details

- Many bugs found and fixed - basic Cthon tests completed
- ACLs were rough - but fruitful discussions and broad agreement on fixes needed
- Need for better implementation documentation for setup :-)
- All great stuff
- Replication/migration and fs_locations missing in action

As an aside

- NFS RDMA prototyping efforts
  - – NFSv3 based
  - – Sun, CITI and NetApp testing and experiments
    - Linux client
    - Sun client and server
    - NetApp server
- Discussions regarding session extensions to NFS as minor revision

NFSv4 Potential Minor Revision Items

- Channel Conjunction Mechanism (CCM)
- Sessions
- Directory Delegation
- Clarifications/bugfixes
- More tbd…

NFS/RDMA Document Review

Document

- “NFS RDMA Problem Statement”
- Tom Talpey and Chet Juszczak
- Updated in February 2004
  - draft-ietf-nfsv4-nfs-rdma-problem-statement-00.txt

Problem Statement

- NFS imposes overhead due to necessary data copies
- RDMA can avoid this overhead
- RDMA solution can be general
- Offloading within the RDMA adapter is additionally beneficial
- User space I/O, etc is enabled

Sources of Overhead

- Diversity of NFS messages
- Variable-length NFS “headers”
- XDR layering and [un]marshalling
- NFSv4 COMPOUND

Unsatisfactory Solutions

- Page flipping / page remapping
- TCP Offload (without NFS support)
- TCP Offload (**with** NFS support)

Quantitative Savings (1)

- Shivam and Chase NICELI paper
- Taking conventional overhead as baseline,
- TCP offload expectation ~25%
- RDMA expectation ~100%

Quantitative Savings (2)

- Callaghan et al NICELI paper
- Conventional NFS (no TOE, with copy) is CPU limited and achieves 60MB/s
- RDMA-enabled NFS not CPU limited up to full 100MB/s GbE bandwidth.

Document

- “NFS RDMA Requirements”
- Brent Callaghan and Mark Wittle
- Published in December 2003
  - draft-callaghan-nfsrdmareq-00.txt

RDMA Requirements

- Basic requirements of an RDMA layer from NFS and RPC
- General to any RDMA-capable transport
- Input to the RDDP Working Group

Basic RDMA Requirements

- Send, RDMA Write, RDMA Read
- Ordered Send completions
- Steering Tags
- Integrity, Privacy desirable

NFS/RDMA Provides

- NFS version support
- Placement information
- Flow control
- Buffer sizing
- Recovery

NFS/RDMA Efficiency

- Operation over IP (RDDP)
- Latency issues
- Transfer size issues

Open Issues from RDDP WG

RDDP IPSec Issues

- What requirements for RDDP/IPSec are desired?
- What is the appropriate level of RDDP IPSec support?
- Should IPSec support be made “mandatory to implement, optional to use” in RDDP, from the NFS/RDMA perspective?

RDDP SSL/TLS Issues

- Is NFS or NFS/RDMA concerned with SSL?
  - SSL over RDMA presents ordering issues
  - RDMA is “loosely ordered”, SSL is not
- What other TLS-style support?
- RPCSEC_GSS interaction?

Parallel NFS (pNFS)

Informational

Open discussion

End