ACL Update

Sam Falkner, Sun
Lisa Week, Sun
Mario Wurzl, EMC

What is the ACL debate?

- The debate boils down to one question: “Why should we change NFSv4 ACL semantics to be like POSIX-draft ACL semantics?”

What does POSIX-draft ACL semantics mean?

- Three things:
  - chmod 644 does not necessarily give read permission to group
  - mode attribute does not show actual group permissions
  - chmod 000 followed by chmod <original-mode> restores the original ACL

Should “chmod 644” always give read permission to group?

- We say “yes”
- RFC 3530 says “yes”
- All existing native implementations say “yes”
- Gruenbacher design says “no”

Should group bits of mode be a superset of all users in the ACL?

- We say “no”
- RFC 3530 says “no”
- Existing native implementations say “no”
- Gruenbacher design says “yes”

Should chmod restore an ACL?

- RFC 3530 doesn't say either way, nor did our minorversion1-06 text
- Every existing native implementation except Sun has said "no"
  - Sun has changed position; Sun no longer wants to restore the ACL
- This leaves only those pushing for a "mask" attribute trying to restore the ACL

Didn’t you guys have a meeting to reach consensus?

- Yes. Consensus reached was to stay with RFC 3530 semantics on definition of group mode bits
- Gruenbacher design violates this decision
  - No requirements justify this change

Any other past issues?

- There was initial argument over whether RFC 3530 “violates POSIX”; all parties now agree that it does not
- Sun’s ZFS algorithms were once normative (but note that there were many optional behaviors) ; as of -06, they are only informative and out of Section 11

What is in minorversion1-06 ACL text?

- Restating original RFC 3530 semantics as requirements (i.e. “SHOULD”, “MUST”, ...)
- Defining access_mask bits; some were ambiguous before
- Clarifying trouble spots (e.g. setting both mode and ACL, DELETE vs. DELETE_CHILD, etc.)

What Now?

- Keep minorversion1-06 ACL section