Unix Extensions for SMB2 Protocol Initiative

Tom Talpey

Microsoft

May 6, 2010

Problem statement

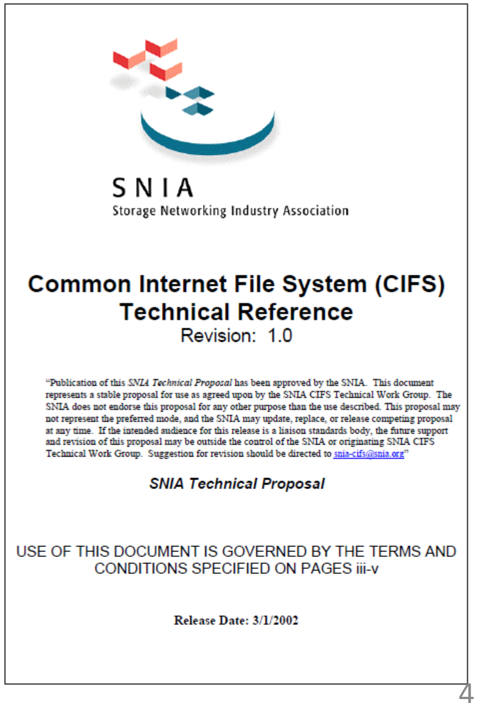
- SMB2 is the file sharing protocol for modern Windows® interoperability
  - Posix semantics are needed for broad deployment of SMB2 on Unix (Linux, Mac, etc)
- Posix is supported only by earlier CIFS/SMB extensions
- Transport encryption not supported by SMB2
- Close these gaps!

Goals

- Enable SMB->SMB2 heterogeneous migration
- Broaden SMB2 interoperability
- Strengthen SMB2 security

Prior work

- SNIA CIFS Technical Working Group
  - http://www.snia.org/tech_activities/CIFS
- Unix and Mac extensions as appendices to SNIA CIFS technical specification
  - Published March 2002
  - (and prior draft versions)



SNIA
Storage Networking Industry Association

Common Internet File System (CIFS)
Technical Reference
Revision: 1.0

"Publication of this *SNIA Technical Proposal* has been approved by the SNIA. This document represents a stable proposal for use as agreed upon by the SNIA CIFS Technical Work Group. The SNIA does not endorse this proposal for any other purpose than the use described. This proposal may not represent the preferred mode, and the SNIA may update, replace, or release competing proposal at any time. If the intended audience for this release is a liaison standards body, the future support and revision of this proposal may be outside the control of the SNIA or originating SNIA CIFS Technical Work Group. Suggestion for revision should be directed to snia-cifs@snia.org"

SNIA Technical Proposal

USE OF THIS DOCUMENT IS GOVERNED BY THE TERMS AND CONDITIONS SPECIFIED ON PAGES iii-v

Release Date: 3/1/2002

Prior approach

- “Carveout” in SMB Information Levels
  - 0x200-0x2ff (only a few actually used)
- Single bit in server capabilities
  - CAP_UNIX negotiation
- Related set of “Macintosh” extensions
  - Similar approach

Problems with previous approach

- “Carveout” problematic
  - Works only between well-matched peers
  - Not readily extensible
  - Led to selective implementation:

    “**12. Appendix D – CIFS UNIX Extension**

    12.1. Introduction

    The purpose of these extensions is to allow UNIX based CIFS clients and servers to exchange information used by UNIX systems, but not present in Windows based CIFS servers or clients. ***These extensions may not be implemented by all UNIX systems***.”
- Limited addressing of needed requirements
  - Overload of existing ops e.g. COM_NT_RENAME
- Not extensible to SMB2

Prior Implementation

- Mixed implementation support
  - Partial, or not implemented on servers
- Minimal common feature interoperation

➢ Clients unable to count on server support

➢ Clients forced to reduce expectations

➢ Clients forced to reduce functionality guarantees

New Protocol Initiative

- A public collaborative effort to create new protocol extensions for the SMB2 protocol, to support Posix/Unix behaviors
- A lightweight, focused process which will draw on prior CIFS/SMB extensions and industry experience
- Will produce an open published specification, available for implementation industry-wide
- Specification(s) to be independent of Microsoft Open Protocols document set
  - MS-SMB2

New Scope and Approach

- Scope:
  - Support Posix
  - Support transport encryption
  - Explicit negotiation
- Approach
  - Explicit reserved Information Levels
  - Explicit reserved ioctls/fsctls, if needed
- Independent of MS-SMB2 protocol spec
  - May make references into other document(s)

Functional Areas

- Protocol changes
  - New capabilities, operations, etc
  - Wire-visible only
- Server best practice
  - Rules and recommendations to ensure interop
- Restrictions
  - Character set, reserved characters, reserved names
- Improvements, protocol relationships, etc

Protocol Specification Scope

- Negotiation of server extension by client (*)
  - Including negotiation of specific supported items below
- Encapsulation of Posix/Unix capabilities:
  - Filename case sensitivity support (*)
  - Posix attributes tunneled: UID/GID, permissions, sizes, times, etc. (*)
  - Posix filename character set support
  - Posix semantics on rename of open files
  - Posix semantics on unlink of files
  - Posix semantics on read-only directories
  - Posix file mode set/get
  - Posix file locking (fcntl(F_GETFL/F_SETFL), advisory vs. mandatory)
  - fcntl(F_FULLFSYNC) (Darwin FUA)
- Transport encryption support

(*) indicates support by prior SMB/Unix extensions

Out of Scope

- Protocol support not explicitly included above
  - Add to scope only with unanimous agreement
- Implementation
  - Producing <u>only</u> a document.

Project Output

- Specification
  - The document itself
- Discussion
  - Exchange of ideas and experience to shape the importance of the specification details
    - What’s most important, what’s less
    - Understand approaches to each
    - Understand comparisons with NFS
- Consensus
  - Community willingness to advance and implement the effort

Project Structure

- “Project Coordinator”
  - Tom Talpey, Microsoft
- Contributors
  - recruiting now!
- Website administration
  - Chris Hertel and José Rivera, ubiqx Consulting

Contributor’s Agreement

- Companies and/or individuals will grant certain rights to their contributions to the Project
- Ensures that use and development of the Specification is available to all
- Final review process (45 days) at project completion
- Click-through acceptance

Commitment from Contributors

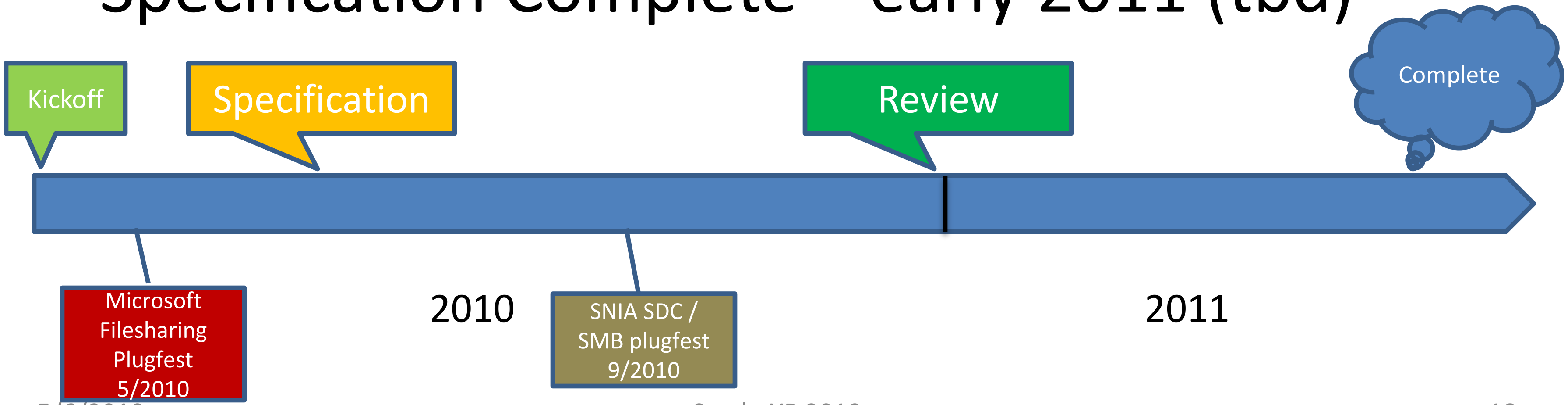
- Willingness to contribute to specification
  - Content (ideas, experience)
  - Writing/editing
  - Reviewing
- Intention to implement is not necessary
  - Though exciting!

Communication

- All discussion to take place via email reflector
- Face-to-face or conference calls possible
- Wiki-style issues list, work-in-progress, etc
- Postings to email and wiki are “Contributions” and may be included in the Specification
- Discussions will be open to public as read-only
- Work-in-progress restricted to Contributors

Schedule

- A 6-12 month effort envisioned
- Project kickoff and recruiting – now
- Project work begins – June 1
- Specification Development – Summer/Fall
- Specification Complete – early 2011 (tbd)


Kickoff
Specification
Review
Complete
Microsoft Filesharing Plugfest 5/2010
2010
SNIA SDC / SMB plugfest 9/2010
2011

Food for Thought - 1

- Unix UID, GID, mode bits
- Symlink support – any incompatible semantics?
- Representing special files (incl devices, pipes, fifo)
- case sensitive, case preserving
- Character set
- Reserved filenames
- ACL interop (subgroup?)
- Locking semantics
  - advisory, stacking, merge/split/overlap, range/offset/signed, behavior on close
  - Windows locking conflicts w/Posix apps
  - Lock state recovery after transport break
- Inode #, other attrs
- Extended attrs

Food for Thought - 2

- Atomicity / replay (multiple operations)
- Return explicit Posix errors from server?
- Rename/delete semantics
- Statfs payload
- Notify semantics w/Posix extensions in effect
- Read and write operation semantics with Posix locking in effect
- Leases
- Extended attributes
- Negotiate capabilities on per-mount basis?
- Posix flush requirement re stat(2)

Food for Thought - 3

- **“Protocols are forever”**
- It’s important to get them right
- IETF model is proven
- Specify, then implement, then decide

Call to Action

- Watch for the website
  - http://unixsmb2.org
- Read the Contributor's Agreement there
- Sign up
- Contribute!

QUESTIONS?

Tom Talpey: ttalpey@microsoft.com

http://unixsmb2.org