NFSv4.1 Client for Windows

Research Project Description

The purpose of this Research Project is to implement an NFSv4.1 client running on Windows.
- This will be built on the Windows Driver Kit (WDK) interfaces, and will borrow learning and some code from University's Unix implementation. It will also be based on the standards specification and interoperable with standards-compliant NFSv4.1 servers.
- A goal of the Research Project will be to achieve interoperability with, or resolve interoperability issues with, key commercial implementations of NFSv4.1 servers, including servers from Sun, NetApp and IBM. During the course of the Project, University will identify and, in collaboration with Microsoft, resolve any ambiguities in the specification and the implementation choices in order to work well on the Windows platform and maintain interoperability with standard-compliant NFSv4.1 servers.
- If any changes to the NFS standards specification are needed, University will take the lead in resolving any issues through the standards process.
- For all Unix/Windows interoperability-related issues encountered during the Research Project, University will create a technical report identifying the issue, and specifying the behavior used by University for the client and server implementations.
- The Research Project will also include development of a Redirected Drive Buffering Subsystem (RDBSS) provider on Windows as well as a module sitting over the Winsock APIs (the "Windows Based Code"). The Windows Based Code will be developed as separate modules from other project deliverables. University will not knowingly use any software for this Windows-Based Code that is subject to a GPL-type license or any IP rights of third parties.
- The current NFSv4.1 specification, draft 26, is in IETF Last Call. Development activities will be directed to the latest draft or RFC for NFSv4.1.

Project Deliverables

As part of this effort, University will develop, release and support the following:
1. An NFSv4.1 client on Windows Vista x64 and Windows Server 2008 x64 platforms using the Windows Redirected Drive Buffering Subsystem (RDBSS) driver and library (http://msdn.microsoft.com/en-us/library/ms794132.aspx) and the Winsock API (http://msdn.microsoft.com/en-us/library/ms740673.aspx).

   The NFSv4.1 client MUST support:
   a. All REQUIRED NFSv4.1 operations, including
      i. Kerberos V5 authentication and Kerberos V5 integrity
      ii. All REQUIRED NFSv4.1 file attributes
      iii. NFSv4.1 ACLs
   b. All REQUIRED pNFS operations
   c. IPv4 and IPv6
   d. Windows based NFS client management tools
   e. The following Optional features:
      i. Referrals
      ii. pNFS file layout
      iii. Byte-range locks
      iv. Kerberos V5 privacy

   The NFSv4.1 client SHOULD support the following Optional features:
   a. All REQUIRED File Delegation operations
   b. Named attributes
   c. Volatile file handles
   d. Mandatory locking

   The NFSv4.1 client MAY support the following Optional features:
   a. All REQUIRED Directory Delegation operations
   b. Blocking locks and polling when server doesn't support the same
   c. Directory caching
   d. Crash recovery of client/server
   e. Client ID trunking support

f. Session trunking support
g. Share reservation support
h. Callback directory notifications
i. Callback recall of delegations
j. Callback to cancel pending delegation wants
k. Callback lock notifications

The NFSv4.1 client MUST NOT support any operations marked MUST NOT IMPLEMENT.

The NFSv4.1 client must be tested against the following test tools:
- Connectathon
- Iozone
- Bonnie++
- Postmark
- BangFS (Stress) Test - http://msdn.microsoft.com/en-us/library/bb961335.aspx
- Syscache (Data Verification) Test - http://msdn.microsoft.com/en-us/library/bb931744.aspx
- Usedisk2 Test - http://msdn.microsoft.com/en-us/library/bb931796.aspx
- FileIO
- Application Verifier and Driver Verifier for all binaries developed with all options enabled

2. Technical reports detailing interoperability issues encountered while implementing the NFSv4.1 client on Windows as specified in Project Deliverables (1) above.

The University will also participate in and report on Connectathon and Bakeathon pNFS interoperability events.

Research Project Term
The Research Project Term will be two (2) years.