NFSv4 Open Source Project Update

Andy Adamson
CITI, University of Michigan
Ann Arbor

A little bit of history

- NFSv4 Open Source Reference Implementation Project
- Sponsored by Sun Microsystems
- IETF reference implementation
- Linux and OpenBSD

Linux NFSv4 Release

- Series of patches for the 2.4.4 kernel using ext2
- Client and server
- First release that is open to the Linux community
- Testing release at Connectathon

Main Patch

- Delegation
- Byte range locking
- Interoperates with all other implementations
- Good performance
- No dependencies on other kernel patches

NFSv4 Utilites

- Export parser
- GSSD
  - Uses our RPCSEC_GSS user level RPC based on MIT
- Mount

RPCSEC_GSS patch

- Depends on Linux kernel cryptoapi patch
- Portions of Kerberos V5 code into Linux kernel
- GSS Mechanism switching code
- Imports GSS context into kernel

RPCSEC_GSS patch

- GSS header verification
- Data integrity and privacy coming soon
- Required NFSv4 Utilities patch
  - MIT Kerberos V5 patch to enable lib/gssapi/mechglue
  - GSSD patch to log on with machine credentials

NFSv3 and RPCSEC_GSS

- Same code base as NFSv4
- Requires GSSD, Linux kernel crypto patch
- Minimal change to NFSv3 mount
- So far, client only
- External RPC interface unchanged
- Passes ‘smoke tests’ against Solaris 2.8 NFSv3 server – will finish testing at Connectathon

NFSv4 ACL patch

- To be released by months end
- Requires Linux kernel ACL patches
  - Included in current Linux kernels
- Maps Linux ext2 POSIX ACL to subset of NFSv4 protocol
- Named attribute functionality not included

NFSv4 - What's Next

- Test, test, test…
- Fully engage Linux community
- Submit code to Linux 2.5.x kernel
- Finish required functionality
  - Named attributes
  - Reboot recovery (started)

SPKM3

- GSS context establishment working
  - Diffie-Hellman key exchange with target x509 certificate verification
- GSS header verification working
  - HMAC-MD5
- GSS data integrity and privacy is close!
  - This will allow LIPKEY to be coded
- Uses Kerberos V5 mechanism switching and context cache
- Uses Openssl crypto

OpenBSD

- Shares code with Linux implementation
  - Most of encode and decode (RPC Compound)
  - Most of the state code (OPEN/LOCK)
- Concentrating on client
  - Passing Connectathon tests
- Release of client at months end

Any questions?

http://www.citi.umich.edu/