Project Status

NFSv4 Extensions for Performance and Interoperability

Center for Information Technology Integration
June 19, 2008

Sessions in the generic Linux pNFS client

Task	Description	Status
S1	Session recovery.	Session recovery is under active development in the NFSv4.1 development community, with Andy Adamson (NetApp) leading the development activity.
S2	Callback channel.	Complete
S3	NFSv4.1 back channel security using machine credentials.	To provide for back channel security, we added support for machine credentials in the SETCLIENTID call. This makes it possible for the callback client to establish a secure channel to the corresponding principal on the callback server. Patches were committed to Linux 2.6.26-rc1. Now we are working on extending the RPC upcall mechanism so that the callback client can acquire appropriate credentials from gssd. Patches were posted to the linux-nfs mailing list last week and are under discussion.
S4	NFSv4.1 back channel security using secret state verifiers.	No progress to report

Other generic pNFS client issues

Task	Description	Status
C1	LAYOUTGET, LAYOUTRETURN, and CB_LAYOUTRECALL.	We have a draft implementation of LAYOUTGET in hand, with improved handling of partial-file layouts to come. A draft implementation of LAYOUTRETURN works for the simple (whole layout) case. We have a general framework and an untested draft implementation of CB_LAYOUTRECALL, with testing still to come.
C2	CB_RECALL_ANY, RECLAIM_COMPLETE, and CB_RECALLABLE_OBJ_AVAIL.	No progress to report. (So far, the NFSv4.1 development community is deferring work on these non-critical elements.)
C3	Integration of block layout requirements into generic client.	This task is under way and ongoing. The main pNFS branch now includes appropriate hooks for the block driver in the write path.

Task	Description	Status
C4	Implement new NFSv4.1 draft 19–21 pNFS features and behavior.	Layout stateid is under active development in the NFSv4.1 development community, with Andy Adamson (NetApp) leading the development activity. Device notification is under active development in the pNFS development community, with Marc Eshel (IBM) leading the development activity. Draft rewrites have simplified this task considerably by eliminating the ADD operation. XDR formats have been worked out and we have an initial implementation of the generic client and server processing code.
C5	Reboot recovery.	Progress has been made by the NFSv4.1 development community, with Andy Adamson leading the development activity. Current work involves repairing the sessions synchronous and asynchronous error recovery. The next step planned is clientid recovery.

Block layout module

Task	Description	Status
B1	Rebase the implementation from block draft 3 to block draft 6.	Work on block draft 6 revisions are complete. Meanwhile the specification has moved forward to draft 9, so this task is ongoing.
B2	Extend the block layout implementation to support large server block sizes.	Support for server block size greater than the page size is roughed in. The pNFS block client and server can now communicate over iSCSI using large block sizes. This task is currently the active focus of development.
B3	Block layout client implementation based on architectural review.	Based on the architectural review, we need to move disk scanning out of the kernel and into user space, but because we have a working kernel implementation, we are deferring this work and using the kernel version to make progress on adding functionality, e.g., Tasks B4 and B5. We will return to this task.
B4	Support for complex volume topologies uses the Linux device mapper (dm). This code needs to be reviewed to meet performance and quality requirements.	We have a working implementation that needs further testing. When we return to task B3, we will revisit this implementation.
B5	Extend the layout cache implementation to support at least two devices.	We have a working implementation that needs further testing. When we return to task B3, we will update this implementation.
B6	Extend the device mapper to support the asynchronous CB_NOTIFY_DEVICEID callback operation.	No progress to report. Block-specific device notification depends on generic device notification (Task C4). We will begin work on this task soon.
B7	The block layout client must implement a timed lease I/O fencing mechanism to insulate against network partition.	No progress to report

PyNFS

Task	Description	Status
P1	Update PyNFS client and server to support new protocol features in the latest drafts.	The PyNFS client and server now support the latest drafts (minorversion1 draft 23 and pnfs-block draft 9).
P2	Enhance the block server implementation to pass full Connectathon tests.	The PyNFS server passes all Connectathon NFSv4 and non-pNFS NFSv4.1 tests except for the large file test.

Milestone summary

The following tasks were projected to be complete by the May 2008 Connectathon.

Task	Description	Status
S1	Session recovery	**Under way**
S2	Callback channel implementation	**Complete**
B1	Block layout draft 6	**Ongoing**
B2	Server block sizes greater then 4 KB	**Under way**
B3	Revisit block layout client implementation based on architectural review	**Deferred**

The following tasks are projected to be complete by the Fall 2008 Bakeathon.

Task	Description	Status
S3	Back channel security using machine credentials	**No progress**
C1	LAYOUTGET, LAYOUTRETURN, and CB_LAYOUTRECALL	**Testing**
C2	CB_RECALL_ANY, RECLAIM_COMPLETE, CB_RECALLABLE_OBJ_AVAIL	**No progress**
P1	PyNFS block client and server support latest drafts	**Complete**
P2	PyNFS block server passes full Connectathon tests	**Nearly complete**

The following tasks are projected to be under way by the Fall 2008 Bakeathon.

Task	Description	Status
C3	Integration of block layout requirements into the generic client	**Under way**
C4	Draft 19–21 pNFS features and behavior	**Under way**
B4	Complex volume topologies	**Testing**
B5	Copy-on-write	**Testing**

The remaining tasks are projected to be complete by the end of the project.

Task	Description	Status
S4	NFSv4.1 back channel security using secret state verifiers	**No progress**
C5	Reboot recovery	**No progress**
B6	CB_NOTIFY_DEVICEID	**No progress**
B7	Timed lease I/O fencing mechanism	**No progress**