NFSv4 - Linux 2.6.2

February 18, 2004

Chapter 1

Introduction

The goal of this report is an evaluation of the state of the art of the NFSv4 Linux implementation. It was written using :

- Mailing lists NFS et NFS-wg
- source code Linux 2.6.2
- Documents (bibliography)

We also have contacted J. Bruce Field[1] and Peter Astrand[2]. We get an idea of the development state of the project making basics tests. This study is based on Linux 2.6.2 - without patches. Since the use of all patches is not possible (basic 'mount' operation fail), we need to study patches one by one. The hight number of patches (65) slowdown the study.

[1] Active NFS client developer. University of Michigan.
[2] Ph.D thesis : *Design and implementation of a Test Suite for NFSv4 servers.*

Chapter 2

Functionality

2.1 Basic operations.

read, write, open file, change attributes... operations work. We can share data without majors problems. Bugs are reported on the mailing list and can be easily produced. We note bigs lacks in robustness and error recovery (server side), and some bugs with SMP.

Patches are yet available to correct some of this problems (I remember we are working on a non patched Linux 2.6.2)

2.2 Delegation, cache and locks operations

Quite all cache and lock operations are yet implemented. Those functions are still under rapid development. It is not interesting in evaluate them while code is not stabilized.

However, delegation operations is not done, and according to Mr Bruce Fields, there is no work planed on this protocol part.

2.3 Security

Authentication under NFSv4 is negotiated and can be done by 4 ways:

1. Kerberos
2. LYPKEY [X.509]
3. SPKM - 3
4. No authentication

For now, only Kerberos identification is possible (mailing lists). We can access to a NFSv4 Linux without identification, but there are problems with rights management (can't copy rwxrwxrwx files, strange IUD/gid...).

The document [principal] say the implantation of LYPKEY/SPKM has started in 2003. A beta support for those protocols is available at the university of Michigan (internal use only) and should be published in the firsts weeks of Mars[1] 2004.

2.4 Windows compatibility.

Compatibility tests made by the University of Michigan between Linux and Windows done "fairly good results" (J. Bruce Field). Actuals Linux clients could not deal correctly with virtual filehandles. So, only some exports kinds are really supported (read only exports, Linux client). Works on client virtual filehandles support is not started nor planed.

2.5 Replication - Migration - Load balancing

The whiter paper [replication] is available. University of Michigan did experimental tests of migrations. There is no planed work for load balancing.

2.6 Tests tools

2.6.1 Black box

One of the classical performances tests tools for files systems is bonnie++. We tried to use it for scalability tests. The bad robustness (and bugs) of NFSv4 don't allow to use this tool correctly.

According to the Linux development process, the actuals patches and bugs corrections will be fully integrated to the Linux 2.6.3/2.6.4 [Andrew Morton realize]. This versions will permit this kinds of tests.

2.6.2 SpecSFS

SpecSFS is the Standard Performance Evaluation Corp.'s benchmark that measures NFS. Actually SpecSFS is in version 3, and it must rewrite to be used with NFSv4.

2.6.3 White[2] box tools tests

The Linux NFSv4 tests suite (pynfs : python NFS) was written to test functionality of NFSv4. It permit a first evaluation of NFSv4 development in April 2002. This tool is under evaluation.

[1] Thanks to Mr. J. Bruce Field, we may have a copy of this work in february.
[2] Peter Astrand terminology. Document [pynfs].

2.7 Administration

Administration of a basic NFSv4 / Linux 2.6.2 without identification is quite easy[3].

Since actual NFSv4 configuration with authentication require kerberos, the administration of NFS is excessively complex : 6 servers must be configured (kerberos, kerberos admin, DNS, idmap, time server, NFSv4). This complexity is essentially caused by kerberos configuration.

An administrative tool is need to create a real commercial tool. In the future this tool must support configuration of not yet done parts of NFSv4, such as migration, load balancing or multi protocols authentication.

[3]In earlier version, we experiment many problems to use NFSv4 without authentication.

Chapter 3

Suggestions

The Linux NFSv4 state of the art is not enough advanced to allow performances testing, but many problems are yet know, patches are available and are under kernel integration.

- Helping stabilization of actual implantation and helping patches integration
- Testing performances (*after* stabilization)
- Starting not done parts of NFSv4 (Delegation) or related projects (load balancing)

Chapter 4

References

Principal http://www.citi.umich.edu/projects/nfsv4/reports/nfsv4_principal.pdf

replication http://www.citi.umich.edu/projects/nfsv4/reports/replication.pdf

partages http://www.citi.umich.edu/projects/nfsv4/reports/statesharing20030513.pdf

pynfs http://www.cendio.se/~peter/thesis/nfsv4-server-testsuite.pdf