Samba and NFS Integration: Can SMB3.11 and pNFS play nicely together?

Steve French
Principal Systems Engineer – Primary Data

Legal Statement

- This work represents the views of the author(s) and does not necessarily reflect the views of Primary Data Corporation
- Linux is a registered trademark of Linus Torvalds.
- Other company, product, and service names may be trademarks or service marks of others.

Who am I?

- Steve French smfrench@gmail.com
- Author and maintainer of Linux cifs vfs (for accessing Samba, Windows and various SMB3/CIFS based NAS appliances)
- Also wrote initial SMB2 kernel client prototype
- Member of the Samba team, coauthor of SNIA CIFS Technical Reference and former SNIA CIFS Working Group chair
- Principal Systems Engineer, Protocols: Primary Data

Outline

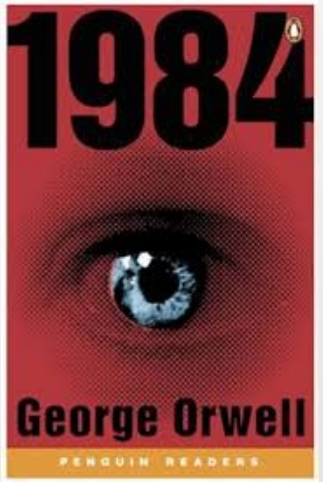
- NFSv4.2 and SMB3: An Overview
- NFS over SMB3
- Can ctdbd help clustered Samba & NFS?
- SMB3 over NFSv4.1/NFSv4.2
- SMB3 and NFS coexistence
  - What about NFSv3?
  - Export of the same data via both protocols over clustered NFS
- Performance overview
- Testing

Why?

- Compatibility, Performance, Stability and Features of Network and Cluster systems vary a lot by OS.
  - SMB3 is very well suited to Windows (and Mac), and particular workloads like HyperV
  - NFS is widely deployed (and understood) for some Linux/Unix workloads. Need to offer NFSv3 and SMB3
- pNFS may be the most common way to access cluster FS

More than one thing happened in 1984

- The two Network File Systems were born in the 80s, devoured their competitors, now reborn stronger ...



NFSv4.x and SMB3.x

- 30 years of improvements have created two quite impressive protocols (NFSv4.2 and SMB3.11)
- One or both are available on very wide variety of clients and servers
  - Although the feature set overlaps each has unique strengths for different workloads
- Well tested
- NFS and SMB are more common than all other network and cluster file systems combined
- Now for a quick review of NFSv4.x and SMB3

Earlier Versions of NFSv4 included

Security

- Uniform namespaces
- Statefulness & Sessions
- Compound operations
- Caching; Directory & File Delegations
- Layouts & pNFS (parallel NFS)
- Trunking (NFSv4.1 & pNFS)

(See e.g. Alex McDonald's talk at http://www.snia.org/sites/default/files/NFS_4.2_Final.pdf)

NFS v4.2 Addresses much more

- Sparse File Support
- Space Reservation
- Labeled NFS
- IO_ADVISE
- Server Side Copy
- Application Data Holes

Many think of stateless, primitive NFSv3 when they think of NFS, or think of NFSv4 or 4.1 but NFSv4.2 adds useful features.

See https://tools.ietf.org/html/draft-ietf-nfsv4-minorversion2-41 for more details

Key NFSv4.2 Feature Status

- Support on client for all major layout types
  - Files (and the follow on flexfiles)
  - Object
  - Block
  - But Kernel Server only has support for block layouts (in 4.0)
- Layoutstats 4.2
- Flexfiles in kernel version 4.0
- Sparse files (3.18 kernel)
- Space reservation (3.19 kernel)
- Labeled NFS (3.11)
- Copy offload
- Not in:
  - Io advise
  - application data holes
  - SCSI file layout

pNFS Variants – “layout types”

- File
  - And now FlexFiles (see presentations by Tom Haynes)
- Object
- Block
- Others have been proposed: Christoph Hellwig's SCSI layout has a draft RFC)

What is Flexfiles?

- Successor to file layout – many improvements
- Aggregation of standalone NFS servers
- Allows reuse of legacy filers as data servers in a clustered configuration
  - Can support legacy NFSv3 data servers
- Exporting of existing clustered file system
  - For example: Ceph, Gluster
- No standard storage access protocol; pNFS could be used instead
- Flexible, per-file striping patterns
- Application SLAs and management policies as well as dynamic load
- balancing and tiering decisions require per-file control over striping
- Existing clustered file systems do not map to the files layout striping patterns

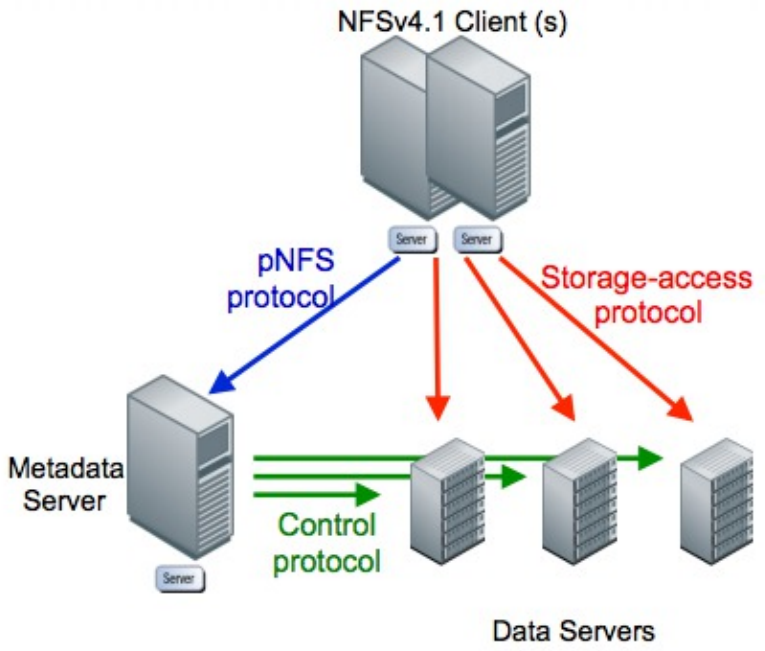
- Allows use of legacy NFSv3 servers for data storage (not just v4.1 and 4.2 servers).
- Allows client side to do the mirroring, offloading some work from the server

Flexfiles

Permit layout to extend over non-pNFS data servers

Example with NFSv3

- File gets private UID and GID that client uses to access the file
- To fence the file: the MDS changes the UID or GID
  - Requires exclusive root access to the data server
- Fences access from clients, and forces clients to:
  - Return the file's layout
  - Request a new layout for the file
- MDS grants access via new UID/ GID to clients it does NOT want to fence
- Only AUTH_SYS is supported to the data servers, not full Kerberos



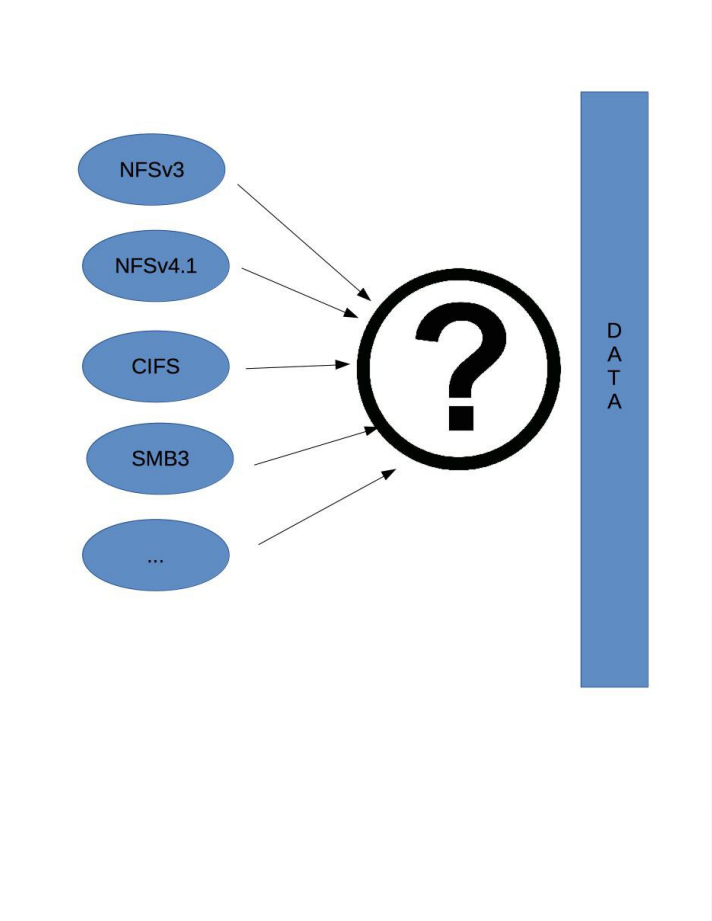
See Alex McDonald's presentation at:
http://www.snia.org/sites/default/files/NFS_4.2_Final.pdf

NFSv4.2 and SMB3.11 comparison

- NFS is more posix compatible
  - e.g. advisory byte range locking and unlink behavior are only emulated on SMB3 on Linux
- PNFS and layout operations are unique to NFS (SMB3 can not separate data and metadata)
- NFS operations are layered over SunRPC (while SMB3 goes directly over TCP) which complicates some optimizations
- Labelled NFS (SELinux security labels)
  - Other xattrs are not as easy to support in NFS although Marc Eshel has draft NFS xattr RFC for this

- SMB3.11 includes a set of loosely related protocols that makes it much broader in scope
  - DFS (Global Name space)
  - Claims Based ACLs
  - File Replication
  - Witness Protocol and unique clustering and HA features
  - User/Group management (and many other admin and management functions)
  - Broader set of file system operations
  - Branch Cache (content addressable)
  - Volume Shadow Copy
  - MS-RSVD “SCSI over SMB3”
  - SMB3 RDMA was developed after NFS RDMA (and added some performance features – making SMB3 RDMA popular)
  - MultiChannel allows better adapter load balancing for SMB3

How to get NFS and SMB3 integration?

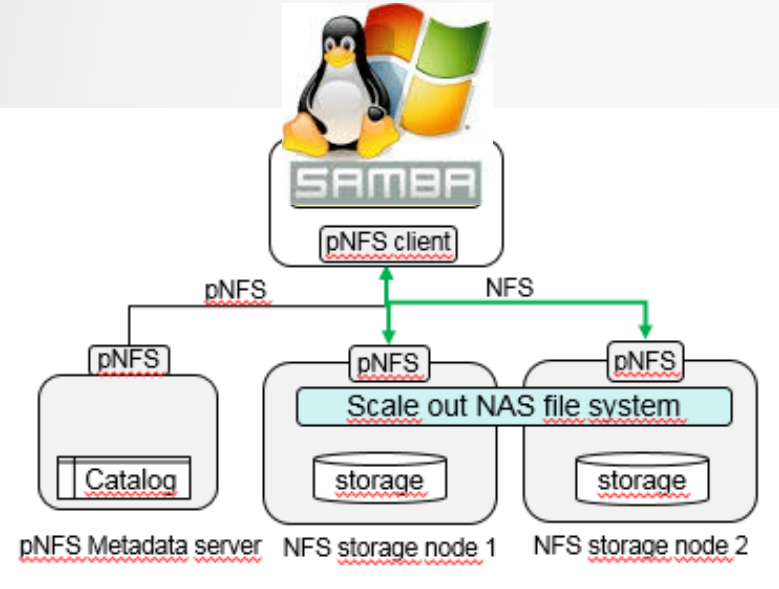


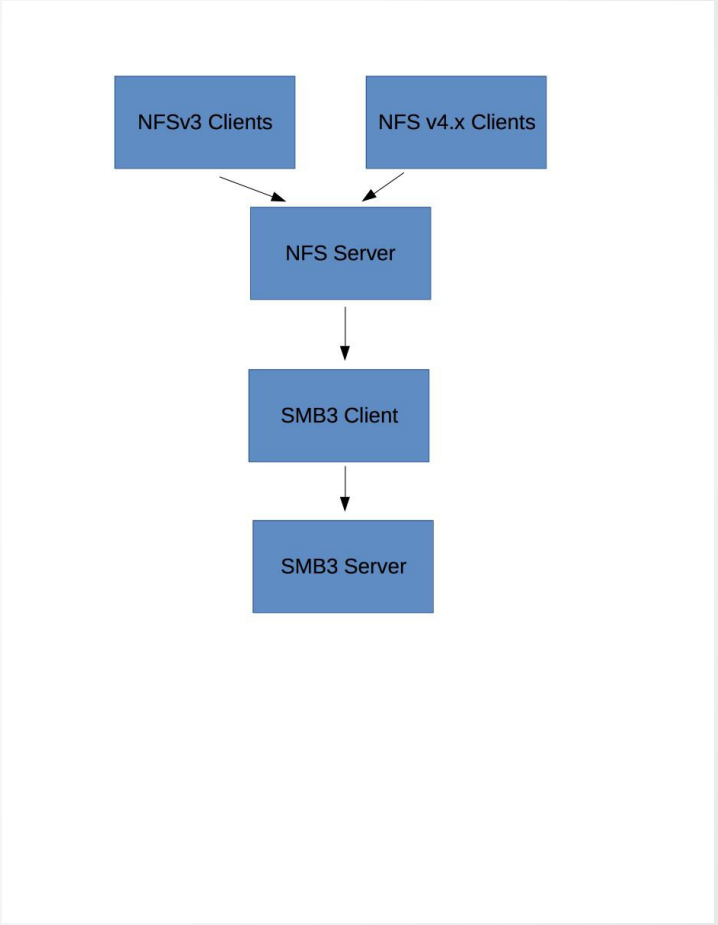
- What is the best way to get NFS and SMB3 … Windows, Mac, Linux … to all access the same data efficiently?

NFS over SMB3 or SMB3 over NFS?

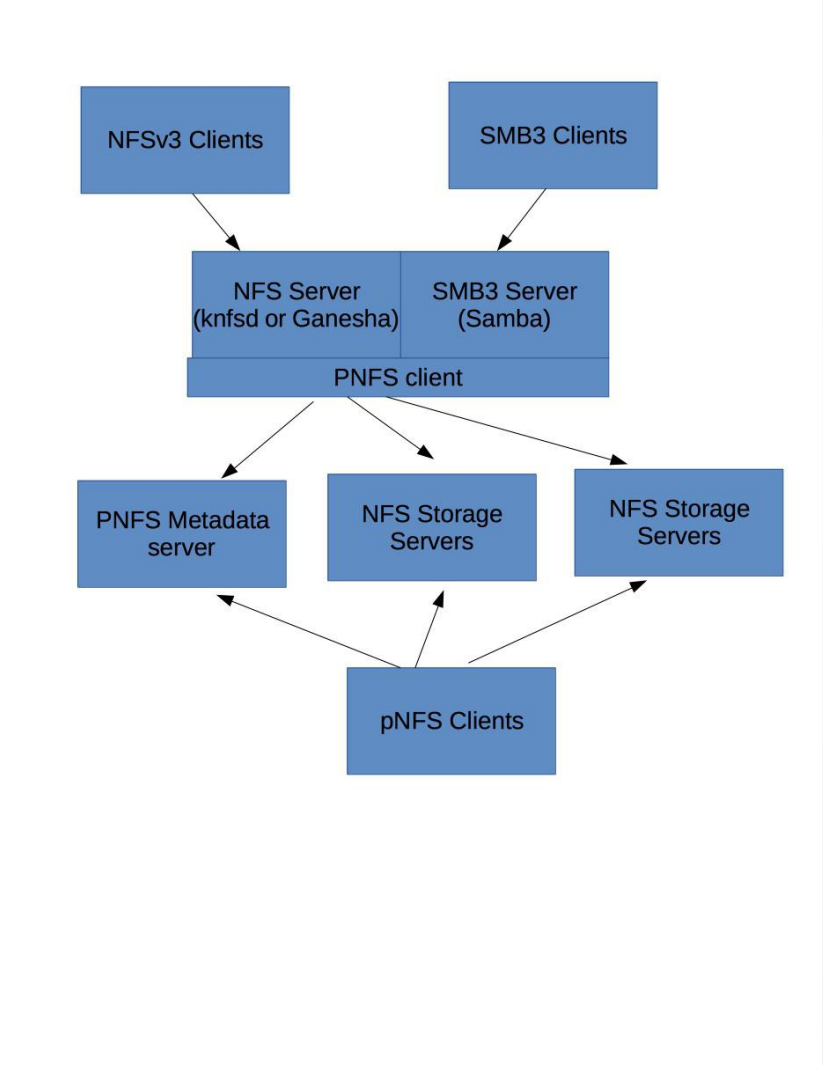
Windows SMB3 Client

Mac SMB3 Client





Dual Gateway over pNFS



- Most likely solution is dual export
  - Samba for Windows, Mac and other SMB3/CIFS clients
  - Knfsd for NFSv3 clients
  - pNFS goes directly to the backend NFS cluster

Key Problems

- Samba uses xattrs to store non-posix inode metadata
  - Creation time
  - DOS attributes
  - Fallback mechanism xattr_tdb (store this metadata in a file)
- ACLs
  - NFS 4.1 (and later, including 4.2) has a similar ACL model to SMB, but lacking “richacl” kernel patches, and NFS richacl enablement, the mapping is incomplete
- Directory Leases
  - NFS does not have directory leases
- Leases/Delegations
  - NFS supports delegations, but not upgrades/downgrades, but the NFS client does not expose these to user space through the vfs api (as the cifs client does)
- Quotas, extra ACL information (auditing information, claims based ACLs e.g.)
- Non-atomic open
  - SMB3 create calls come in with “create contexts” and metadata that can not really be set atomically
- Advisory (NFS) vs. Mandatory (SMB) byte range locking
- No concept of “reparse points” (Samba already has to deal with compensations for this for normal posix fs) or DFS (global namespace) in Linux NFS.
- NFS layouts can not be exposed (unless an SMB ioctl were added) to SMB clients. pNFS I/o has to go through the same Samba server and NFS client under it, rather than letting SMB3 client distribute the file I/o as e.g for apps running directly on flexfiles or files layout pnfs client)
- Mapping security identities
- HyperV related protocol features, QoS, Sparse files, Compressed files ...

SMB3 and NFSv4.x Security Features

- SMB3.1.1 secure negotiate
- SMB3 Share Encryption
- UID/GID mapping and winbind
  - How to get consistent uid (to username, and to SID) across all file servers and the cluster

What about knfsd (or Ganesha) and Samba exporting the same data?

- Need a good cluster file system
- Need something like ctdb to handle starting/stopping services and help Samba with additional non-posix state that the cluster fs can't handle

CTDB and nfs

- CTDB can do a lot for Samba but not as tightly integrated into Ganesh or KNFSD
- Provides some benefits though (see ctdbd config files e.g.):

```
1 # Options to ctdbd, read by ctdbd_wrapper(1). See ctdbd.conf(5) for more information about CTDB configuration variables
5 # Shared recovery lock file to avoid split brain.  No default.
9 # CTDB_RECOVERY_LOCK=/some/place/on/shared/storage
11 # List of nodes in the cluster.  Default is below.
12 # CTDB_NODES=/etc/ctdb/nodes
14 # List of public addresses for providing NAS services.  No default.
15 # CTDB_PUBLIC_ADDRESSES=/etc/ctdb/public_addresses
17 # What services should CTDB manage?  Default is none.
18 CTDB_MANAGES_SAMBA=yes
19 CTDB_MANAGES_WINBIND=yes
20 # CTDB_MANAGES_NFS=yes
21
```

- ctdb can:
  - Start/stop NFS, do some ip address management
  - (Ganesha) help manage nfs grace period to improve reconnect
  - Includes about 15 distinct nfs helper/event scripts, and 40+ collections of small tests for nfs/ctdb events
- Also see:
  - https://wiki.samba.org/index.php/CTDB_Setup#Setting_up_CTDB_for_clustered_NFS
  - https://ctdb.samba.org/nfs.html
  - ctdb/config/nfs-checks.d/README

Knfsd over cifs.ko SMB3 mounts

- Should be possible, but needs some work. See filesystems/documentation/nfs/exporting (has been done but needs to be optionally enabled at least smb3)

```
[root@xfstest Downloads]# mount -t nfs localhost:/public /mnt2 -o vers=4.1
mount.nfs: mounting localhost:/public failed, reason given by server: No such file or di
rectory
[root@xfstest Downloads]# mount -t nfs localhost:/public /mnt2 -o vers=3
mount.nfs: access denied by server while mounting localhost:/public
[root@xfstest Downloads]# cat /proc/mounts | grep cifs
//localhost/portal-share /public cifs rw,relatime,vers=1.0,cache=strict,username=sfrench
,domain=XFSTEST,uid=0,noforceuid,gid=0,noforcegid,addr=127.0.0.1,unix,posixpaths,serveri
no,mapposix,acl,rsize=1048576,wsize=65536,echo_interval=60,actimeo=1 0 0
```

Knfsd over cifs.ko - configuration

- Enable cifs export operations in make menuconfig (see fs/cifs/Kconfig)

  config CIFS_NFSD_EXPORT

      bool "Allow nfsd to export CIFS file system"

      depends on CIFS && BROKEN

    help

      Allows NFS server to export a CIFS mounted share (nfsd over cifs)

Export operations are needed by knfsd

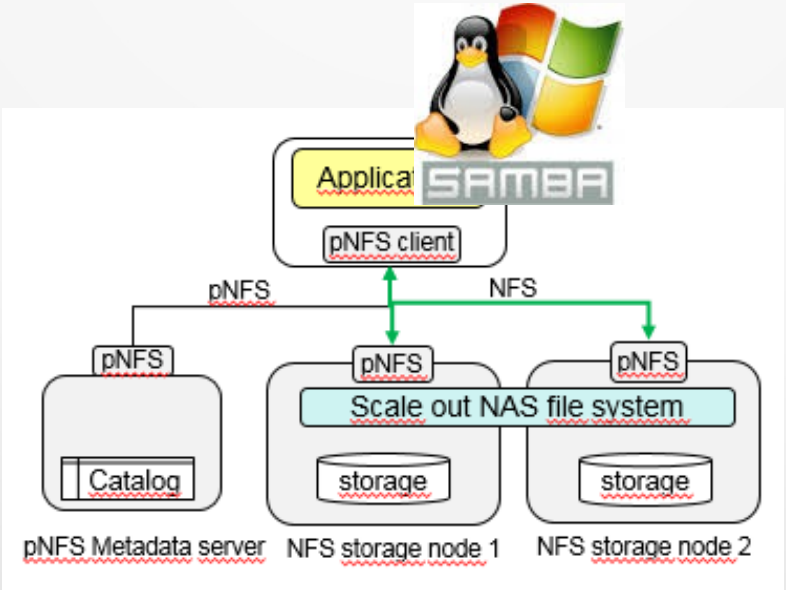
- e.g. struct export_operations btrfs_export_ops

```
.encode_fh  = btrfs_encode_fh,
.fh_to_dentry    = btrfs_fh_to_dentry,
.fh_to_parent    = btrfs_fh_to_parent,
.get_parent  = btrfs_get_parent,
.get_name   = btrfs_get_name,
```

- Cifs.ko has partial implementation

What about knfsd (or Ganesha) and Samba exporting the same data?

- **Need a good cluster file system**
- What if knfsd (or Ganesha) AND Samba run over pNFS?
  - NFSv3 and NFSv4.0 can go to knfsd/Ganesha
  - CIFS/SMB3 can go to Samba
  - pNFS aware clients can go directly to underlying cluster



Samba over pNFS

- Samba (oversimplified) emulation strategy for metadata
  - If it is in posix (e.g. mtime, file size) use it …
    - Else if file system specific vfs module can ... return it
    - Else try xattrs
      - If xattrs enabled get/set the metadata via xattr
    - Else emulate xattrs (e.g. vfs xattr_tdb)
    - Else guess or emulate other ways
      - e.g. “archive” attribute is often mapped to “group execute” mode bit (smb.conf setting “map_archive”)
- But NFS client on linux doesn't support xattrs … (except for posix acls, and security labels) ...

xstat (now “statx”)

- Broad agreement on it at Linux File System Summit in the Spring
- Patches updated and reproposed:
  - See https://lwn.net/Articles/685519/
- Last activity on patches in late May
- Last comment on fsdevel mailine list (Dave Howells/Christoph)

  > And to get back to stat: if would be really useful to coordinate

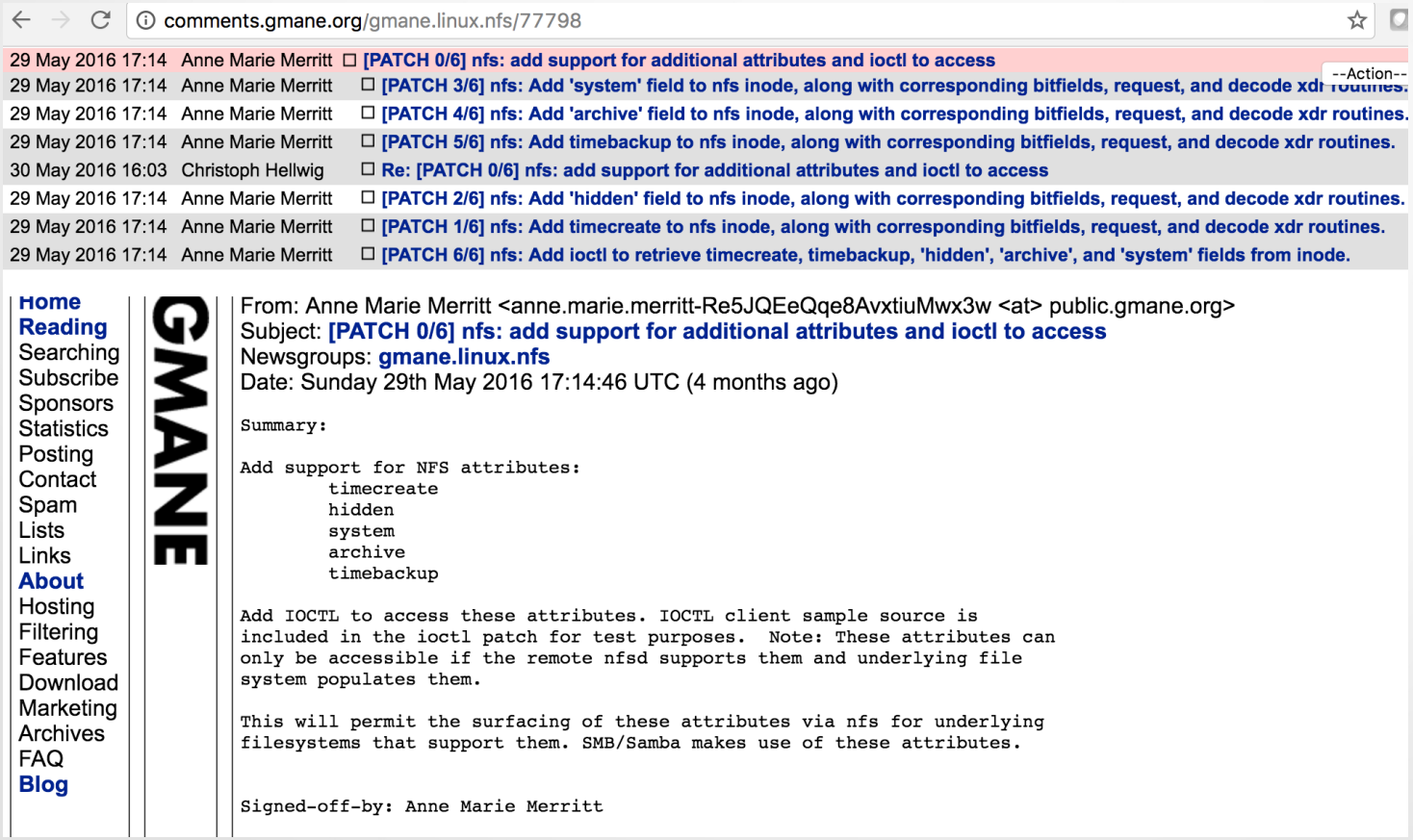
  > new one with glibc so that we don't end up with two different stat

  > structures again like we do for a lot of platforms at the moment.

  I've tried reaching out to them and others, but no one responded.

Creation Time and DOS Attributes

- Patches to nfs.ko proposed to solve
- Christoph: use xstat instead to broaden to more fs



comments.gmane.org/gmane.linux.nfs/77798

29 May 2016 17:14	Anne Marie Merritt	[PATCH 0/6] nfs: add support for additional attributes and ioctl to access
29 May 2016 17:14	Anne Marie Merritt	[PATCH 3/6] nfs: Add 'system' field to nfs inode, along with corresponding bitfields, request, and decode xdr routines.
29 May 2016 17:14	Anne Marie Merritt	[PATCH 4/6] nfs: Add 'archive' field to nfs inode, along with corresponding bitfields, request, and decode xdr routines.
29 May 2016 17:14	Anne Marie Merritt	[PATCH 5/6] nfs: Add timebackup to nfs inode, along with corresponding bitfields, request, and decode xdr routines.
30 May 2016 16:03	Christoph Hellwig	Re: [PATCH 0/6] nfs: add support for additional attributes and ioctl to access
29 May 2016 17:14	Anne Marie Merritt	[PATCH 2/6] nfs: Add 'hidden' field to nfs inode, along with corresponding bitfields, request, and decode xdr routines.
29 May 2016 17:14	Anne Marie Merritt	[PATCH 1/6] nfs: Add timecreate to nfs inode, along with corresponding bitfields, request, and decode xdr routines.
29 May 2016 17:14	Anne Marie Merritt	[PATCH 6/6] nfs: Add ioctl to retrieve timecreate, timebackup, 'hidden', 'archive', and 'system' fields from inode.

--Action--

Home
Reading
Searching
Subscribe
Sponsors
Statistics
Posting
Contact
Spam
Lists
Links
About
Hosting
Filtering
Features
Download
Marketing
Archives
FAQ
Blog

GMANE

From: Anne Marie Merritt <anne.marie.merritt-Re5JQEeQqe8AvxtiuMwx3w <at> public.gmane.org>
Subject: [PATCH 0/6] nfs: add support for additional attributes and ioctl to access
Newsgroups: gmane.linux.nfs
Date: Sunday 29th May 2016 17:14:46 UTC (4 months ago)

```
Summary:

Add support for NFS attributes:
        timecreate
        hidden
        system
        archive
        timebackup

Add IOCTL to access these attributes. IOCTL client sample source is
included in the ioctl patch for test purposes.  Note: These attributes can
only be accessible if the remote nfsd supports them and underlying file
system populates them.

This will permit the surfacing of these attributes via nfs for underlying
filesystems that support them. SMB/Samba makes use of these attributes.


Signed-off-by: Anne Marie Merritt
```

Many SMB3 Attributes are mappable

- /* File Attribute flags*/
- #define ATTR_READONLY 0x0001
- #define ATTR_HIDDEN   0x0002
- #define ATTR_SYSTEM   0x0004
- #define ATTR_VOLUME   0x0008
- #define ATTR_DIRECTORY 0x0010
- #define ATTR_ARCHIVE   0x0020
- #define ATTR_DEVICE   0x0040
- #define ATTR_NORMAL   0x0080
- #define ATTR_TEMPORARY 0x0100
- #define ATTR_SPARSE   0x0200
- #define ATTR_REPARSE   0x0400
- #define ATTR_COMPRESSED 0x0800
- #define ATTR_OFFLINE   0x1000/* ie file not immediately available - on offline storage */
- #define ATTR_NOT_CONTENT_INDEXED 0x2000
- #define ATTR_ENCRYPTED 0x4000
- #define ATTR_POSIX_SEMANTICS 0x01000000
- #define ATTR_BACKUP_SEMANTICS 0x02000000
- #define ATTR_DELETE_ON_CLOSE 0x04000000
- #define ATTR_SEQUENTIAL_SCAN 0x08000000
- #define ATTR_RANDOM_ACCESS   0x10000000
- #define ATTR_NO_BUFFERING   0x20000000
- #define ATTR_WRITE_THROUGH   0x80000000

SMB3, NFSv4.x and ACLs

- The ACL model is surprisingly close between NFSv4.1 and SMB3 (if don't include claims based ACLs which are unique to SMB3, and Apache etc. but not in NFS spec)
  - username@domain (nfsv4.x) usually maps 1 to 1 to SIDs (smb3)
- RichACLs would make this so much easier
  - At the file system summit this spring, its outlook looked promising
  - Last update was about a month ago, expect more activity for 4.9
    - https://lwn.net/Articles/695289/
    - http://marc.info/?l=linux-fsdevel&m=147134547318959&w=2
- Deny bits reordering is an issue
  - Mode bits are often represented with deny aces out of order in richacl (Windows tools prefer deny ACEs first)
  - See https://blogs.msdn.microsoft.com/oldnewthing/20070608-00/?p=26503

Other Optional features

- Xstat integration
  - Returns birth time and dos attributes in more standardized fashion (cifs has a private xattr for that, but few tools use it). Kernel patches exist, would help cifs a lot
- alternate data streams
- Clustering, Witness protocol integration
- DFS
- Other ...

Key Samba activity

- Merge richacl into kernel and turn on vfs_richacl
  - NFS specific Samba VFS ACL module is out of date, and less useful
- vfs_xattr_tdb
- vfs_ … to enable clone/copy range and fallocate funcxtions
- xstat enablement or equivalently vfs_nfs… to call nfsioctl for attributes, and creation time (using something like Anne-Marie's patch set in nfs.ko)

SMB3 and Performance

- Key Features
  - Compounding
  - Large file I/O
  - File Leases
    - Lease upgrades
  - Directory Leases
  - Copy Offload
  - Multi-Channel
    - And optional RDMA
  - Linux specific protocol optimizations
- How are these affected in a gateway like environment?

SMB3 over pNFS/NFSv4.2 Performance Overview

- Obviously more traffic but two promising things
  - NFSv4.2 support for clone/copy offload

    #ifdef CONFIG_NFS_V4_2

    .copy_file_range = nfs4_copy_file_range,

    .llseek = nfs4_file_llseek,

    .fallocate = nfs42_fallocate,

    .clone_file_range = nfs42_clone_file_range
  - NFS client asks for delegations on the wire (but the client doesn't expose them, which could be fixed, as cifs.ko did to allow setlease and break lease to work if delegation already held)
    - NFS does not support delegation upgrades but this is not as important as basic case of getting lease on typical uncontested file
    - NFS does not support directory delegations but can cache directory metadata loosely

Testing … testing … testing

- How to test
  - NFS to knfsd (or Ganesha) over pNFS
  - SMB3 to Samba over pNFS
  - Multiple protocols at the same time exported over same underlying pNFS cluster
- Xstat works over BOTH cifs and nfs clients
- Smbtorture (to Samba)
- PyNFS - to NFS server(s)
- Cross-protocol tests could use improvements but dbench and others can be used for simultaneous configurable load from multiple protocols

Thank you for your time