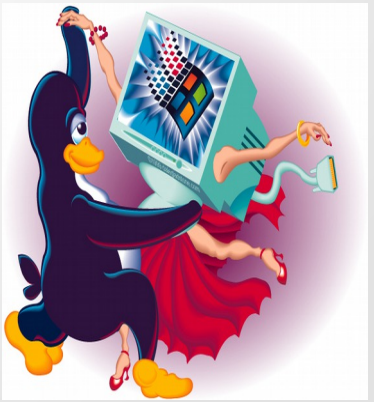
Pushing the Boundaries of SMB3: Status of the Linux Kernel client and interoperability with Samba

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

- File System Activity
- Key Feature Status
  - Completed Features
  - In progress features
  - Other optional SMB3 features
- Performance overview
- POSIX compatibility and extensions
- Testing

A year ago … and now … kernel (including cifs client) improving

- 12 months ago we had Linux version 4.6-rc6 ie “Charred Weasel”

Two days ago we got 4.11-“Fearless Coyote”

Working with great developers. Here we are at 2017 Linux File System Summit in Cambridge

Some key features helping drive discussions and FS development activity ?

- Many of high priority, evolving storage features are critical for NAS
  - Better support for NVMe
    - RDMA, low latency ways to access VERY high speed storage
  - Faster (and more) network interfaces
  - Security/crypto Improvements
  - RichACL (maybe someday … we can hope ...)
  - statx (extended stat)
  - Improved copy offload
  - Improved sparse file support (including for virtualization)
  - Shift of some workloads to object like access patterns

Most Active Linux Filesystems this year

- 4352 kernel filesystem changesets in last year (since 4.6-rc5 kernel)!
  - Linux kernel file system activity continuing strong (although down about 2% due to gradual maturing)
  - FS activity accounts for 5.8% of overall kernel changes (which are dominated by drivers) but fs is watched carefully, fs activity slightly higher percentage of kernel than last year
  - Kernel is now > 15.6 million lines of source code (measured last week with sloccount tool)
- There are many Linux file systems, but seven (and the VFS layer itself) drive the majority of activity
  - File systems represent about 5.5% of the overall kernel source code (859,000 lines of code)
- cifs.ko (cifs/smb3 client) among more active fs
  - Btrfs 735 changesets (down about 5%)
  - VFS (overall fs mapping layer and common functions) 689 (increased)
  - XFS 524 (up slightly)
  - Nfs client 452 (down slightly)
  - Ext4 259 (up about 7%)
  - Ceph 214
  - CIFS/SMB2/SMB3 client 180 (up about 30%)
    - cifs.ko is 43,570 lines of kernel code (not counting user space helpers, and samba userspace tools)
  - Nfs server 150 (flat)
- NB: Samba (cifs/smb2/smb3 server) is as active as the top 3 or 4 put together (thousands of changesets) since it is broader in scope (by a lot) and also is in user space not in kernel

Fixes and Features by release

- Linux 4.2 (14 changesets)
  - SMB 3.11 (Windows 10) dialect support (improved security)
  - Faster copy offload (REFLINK, duplicate_extents) added for Windows Server 2016
- 4.3 (17 changesets)
  - Minor bug fixes (including Mac authentication issue when timestamps differ too much on server/client)
  - Add krb5 support for smb3
  - cifs.ko version updated to 2.08
  - Added ioctl to query detailed fs info on mounted share
- Linux 4.4 (17 changesets)
  - Allow copy offload across shares
  - Add resilient and persistent handle mount options and support for the (durable v2) create context

Fixes and Features (continued)

- Linux 4.5 (27 changesets)
  - Minor bug fixes
  - clone_file_range added to vfs, cifs support for clone_file_range
  - Allow O_DIRECT with cache=loose
  - Make echo interval tunable
  - (first phase of encryption support begun)
- Linux 4.6 (8 changesets)
  - Minor fixes
- Linux 4.7 (7 changes)
  - Fix badlock regression for guest mounts (mount with -o guest can fail to Samba servers when patched for badlock)
  - Cifs.ko version updated to 2.09
  - Minor fixes: including NetApp DFSpathname issue, Improved reconnection support and POSIX pathname and special character (trailing colon and space)
- 4.8 (18 changesets)
  - Allow mounts with prefixpath where top of share unaccessible
  - Fix for create when existing directory of same name\
  - mfsymlink support added for smb2/smb3 (symlink emulation, also used by Mac)
  - Misc minor fixes

Fixes and Features (continued)

- 4.9 (37 changesets)
  - Various reconnect improvements (e.g. send echo ASAP to reconnect smb session/tcon quicker after socket reconnect)
  - Uid/gid from special sid (new mount option “idsfromsid”)
  - Can override number of credits (new mount option “max_credits”)
  - Query file attributes or creation time via xattr
- 4.10 (17)
  - New snapshot mount parm (“snapshot”)
  - Misc bug fixes
- 4.11 (51 changesets)
  - SMB3 reconnect improvements (including better persistent & durable handles). Much higher reliability now when server crashes or failsover while I/o in flight or cached. Lots of corner cases fixed (Thank you Germano!)
  - Server side copy works much better: Clone file range (and “cp –reflink” command) now support more common “copychunk” copy offload style (had required ess common “duplicate extents” support). Thank you Sachin!
  - SMB3 DFS support (Thank you Aurelien – see his presentation!)
  - SMB3 Encryption support (Thank you Pavel!)

Fixes and Features (continued)

- 4.12-rc (7 changesets + …)
  - Blazingly fast kernel AIO (performance improvement)
  - Bug fixes (including prefixpath matching on 2nd mount, copy offload fix)
  - (coming soon) statx, smb3 acls
- Goals for 4.13
  - RDMA !
  - POSIX Extensions for SMB3 (even if experimental)

Fixes and Features in progress – What we talked about at SDC – almost there!

- ~~Prefix path fixes~~
- ~~Improved POSIX compatibility~~ (~~some~~ work in progress e.g. SMB3 POSIX Extensions)
- ~~Return important SMB3 inode metadata via xattrs (create time, attributes, ADS names)~~
- ~~Improved reconnect and HA support~~
- ~~Encrypted Share support~~
- ACLs and security improvements (in progress)

What are most noticeable, most important improvements over last year?

- IMO (Other opinions welcome, lots of candidates)
  - Much better reconnection after server or network failure, data integrity (not just persistent handles)
  - SMB Encryption
  - Performance:
    - Copy offload (“cp –reflink”) super fast to Samba, Windows etc.
    - AIO much faster
  - SMB3 DFS (Global Name Space Support)
  - Snapshot support
  - And of course SMB3 continues to improve ...

statx()

- After multiple years of technical discussion it is now merged into Linux kernel!
- VFS support and system call added In 4.11 kernel
  - CIFS enablement planned for 4.12
  - Can return creation time and a few new attributes (including e.g. 'compressed')
  - Extensible, more flags coming (for query more metadata that Samba and cifs.ko care about)
  - 'set' for statx also planned to be added into the vfs (then in cifs) but this first step is important

Linux CIFS/SMB3 client bug status summary

- See bugzilla.kernel.org cifsvfs component
  - Fewer than 50 bugs
  - Most not serious (or fixed) on 4.11
  - Also see bugzilla.samba.org

SMB3 Capabilities supported

- SMB2 CAP DFS
- SMB2 CAP LEASING
- SMB2 CAP LARGE_MTU
- **SMB2 CAP PERSISTENT HANDLES**
  - Client support added in Linux kernel 4.4, reconnection much improved in 4.10
  - (NB: Samba server support is in progress)
- **SMB2 CAP ENCRYPTION**
- Unsupported capabilities
  - SMB2 CAP DIRECTORY LEASING
  - SMB2 CAP MULTI CHANNEL (not in client, though is supported in Samba server since Samba server version 4.4)

SMB3 AIO is so fast now!

- (Thank you Pavel!!)
- Size 4KB, sequential reading/writing. Virtual Machine with 4 cores, 32GB RAM (large enough so reads satisfied from cache), HDD
  - IO depth 1 98MB/s read (same as before AIO patches)
  - IO depth 32 450 MB/sec reading, 140 MB/s writing (much faster than before AIO patches)
  - Note the 450% improvement in read (mostly cached) with these patches, and even writing > 40% faster
- But what if we used faster storage (SSDs) … and how would that compare to NFS?

Using SSDs instead - AIO flies!

- With AIO patches, current cifs.ko (SMB3):
  - I/O depth 1, 98MB/s read, 85 MB/s write
  - I/O depth 32 450 MB/s read (!), 320 MB/s write (!)
  - Increasing I/O depth to 64 or higher did not get any faster with this speed disk but might in other hardware configurations
- Without AIO patches, previous cifs.ko (SMB3)
  - Much slower: Results are similar to I/O depth 1
  - AIO patches: 4.5 times faster read! 3.8 times faster write!
- What about NFS?
  - I/O depth 1: 85 MB/s reading, 65 MB/s writing
  - I/O depth 32: 200 MB/s reading, 145 MB/s writing
- SMB3 with AIO patches Is more than twice as fast as NFS with same I/O depth (even today it is much faster than nfs without these patches for I/o depth 1)
- Looks fantastic!

Copy Offload – big performance win

```
root@ubuntu:~# dd if=/dev/zero of=/mnt1/30M count=300 bs=100K
300+0 records in
300+0 records out
30720000 bytes (31 MB) copied, 0.445072 s, 69.0 MB/s
root@ubuntu:~# ls /mnt1
30M  3M  copy-of-3M  normal-non-ss-copy-of-3M  public
root@ubuntu:~# rm /mnt1/copy-of-3M
root@ubuntu:~# rm /mnt1/normal-non-ss-copy-of-3M
root@ubuntu:~# time cp /mnt1/3M /mnt1/normal-non-ss-copy-of-3M

real    0m0.068s
user    0m0.000s
sys     0m0.032s
root@ubuntu:~# time cp /mnt1/30M /mnt1/normal-non-ss-copy-of-30M

real    0m0.484s
user    0m0.000s
sys     0m0.351s
root@ubuntu:~# time cp --reflink /mnt1/3M /mnt1/ss-copy-of-3M

real    0m0.018s
user    0m0.000s
sys     0m0.007s
root@ubuntu:~# time cp --reflink /mnt1/30M /mnt1/ss-copy-of-30M

real    0m0.020s
user    0m0.000s
sys     0m0.010s
root@ubuntu:~#
```

DUPLICATE_EXTENTS is very efficient

```
    520 8.758876000 192.168.93.136   192.168.93.130  SMB2        342 Create Request File: ss-copy3-of-30M
    521 8.759457000 192.168.93.130   192.168.93.136  SMB2        334 Create Response File: ss-copy3-of-30M
    522 8.759611000 192.168.93.136   192.168.93.130  SMB2        175 GetInfo Request FILE_INFO/SMB2_FILE_INTERNAL_INFO File: ss-copy3-of-30M
    523 8.759911000 192.168.93.130   192.168.93.136  SMB2        150 GetInfo Response
    526 8.760144000 192.168.93.136   192.168.93.130  SMB2        191 Ioctl Request FILE_SYSTEM Function:0x0031 File: ss-copy3-of-30M
    527 8.760487000 192.168.93.130   192.168.93.136  SMB2        182 Ioctl Response FILE_SYSTEM Function:0x0031 File: ss-copy3-of-30M
                    192.168.93.136   192.168.93.130  SMB2        174 SetInfo Request FILE_INFO/SMB2_FILE_ENDOFFILE_INFO File: ss-copy3-of-30M
                    192.168.93.130   192.168.93.136  SMB2        136 SetInfo Response
    530 8.761086000 192.168.93.136   192.168.93.130  SMB2        230 Ioctl Request FILE_SYSTEM Function:0x00d1 File: ss-copy3-of-30M
    531 8.761481000 192.168.93.130   192.168.93.136  SMB2        182 Ioctl Response FILE_SYSTEM Function:0x00d1 File: ss-copy3-of-30M
    532 8.761610000 192.168.93.136   192.168.93.130  SMB2        158 Close Request File: ss-copy3-of-30M
    533 8.767873000 192.168.93.130   192.168.93.136  SMB2        194 Close Response
```

LibreOffice Impress

```
▶ Frame 530: 230 bytes on wire (1840 bits), 230 bytes captured (1840 bits) on interface 0
▶ Ethernet II, Src: Vmware_b4:dc:f2 (00:0c:29:b4:dc:f2), Dst: Vmware_84:48:c0 (00:0c:29:84:48:c0)
▶ Internet Protocol Version 4, Src: 192.168.93.136 (192.168.93.136), Dst: 192.168.93.130 (192.168.93.130)
▶ Transmission Control Protocol, Src Port: 41774 (41774), Dst Port: microsoft-ds (445), Seq: 1469, Ack: 1432, Len: 164
▶ NetBIOS Session Service
▼ SMB2 (Server Message Block Protocol version 2)
  ▶ SMB2 Header
  ▼ Ioctl Request (0x0b)
    ▶ StructureSize: 0x0039
    ▶ Function: Unknown (0x00098344)
    ▶ GUID handle File: ss-copy3-of-30M
      Max Ioctl In Size: 0
      Max Ioctl Out Size: 65280
    ▶ Flags: 0x00000001
```

Duplicate Extents vs CopyChunk for server side copy (to REFS)

```
root@ubuntu:~/xfstests-new/xfstests-dev# dd if=/dev/zero of=/mnt1/500M count=500 bs=1M
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 17.212 s, 30.5 MB/s
root@ubuntu:~/xfstests-new/xfstests-dev# time cp /mnt1/500M /mnt1/normal-copy-500M

real    0m19.972s
user    0m0.004s
sys     0m0.289s
root@ubuntu:~/xfstests-new/xfstests-dev# ./src/cloner /mnt1/500M /mnt1/copy-chunk-500M
root@ubuntu:~/xfstests-new/xfstests-dev# time ./src/cloner /mnt1/500M /mnt1/copy-chunk-500M

real    0m0.531s
user    0m0.000s
sys     0m0.061s
root@ubuntu:~/xfstests-new/xfstests-dev# time ./src/cloner /mnt1/500M /mnt1/copy-chunk-500M-try2

real    0m18.513s
user    0m0.000s
sys     0m0.075s
root@ubuntu:~/xfstests-new/xfstests-dev# time cp --reflink /mnt1/500M /mnt1/reflink-copy-500M

real    0m0.034s
user    0m0.000s
sys     0m0.009s
root@ubuntu:~/xfstests-new/xfstests-dev#
```

Amazon

Fallocate (works, but minor TODOs)

- We currently support
  - Simple fallocate
  - PUNCH_HOLE
  - ZERO_RANGE
  - KEEP_SIZE
- We have discussed ways to add support for the remaining two when the server supports duplicate extents (currently REFS on Windows 2016 is the only one that advertises “FS_SUPPORTS_BLOCK_REFCOUNTING” capability). We can add support for:
  - COLLAPSE_RANGE
  - INSERT_RANGE

SMB3 Security features (TODOs)

- Finish up SMB3.1.1 secure negotiate
- Finish up ACL query

SMB3 and Performance (focus areas **highlighted,** already supported areas normal text)

- Key Features
  - Async and vectored I/O
  - **Compounding (reduce number of roundtrips)**
  - Large file I/O
  - File Leases
    - **Lease upgrades**
  - **Directory Leases**
  - Copy Offload
  - **Multi-Channel**
    - **And optional RDMA**
  - Linux specific protocol optimizations

SMB3 POSIX Compatibility

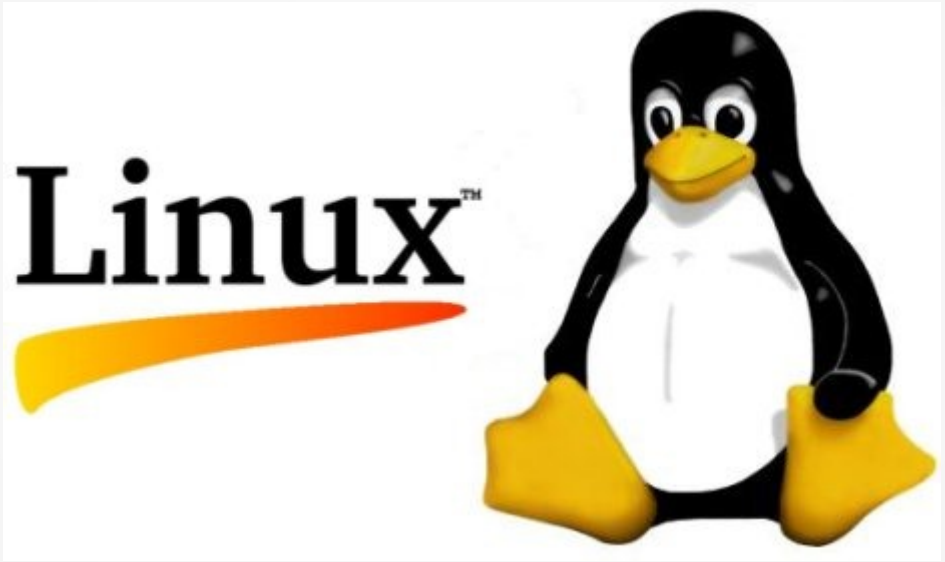
- CIFS has good posix compatibility with Samba, but we want to disable old cifs, and for security and performance and better features only use SMB3 and later
  - But SMB3 didn't have POSIX extensions to the protocol
- What does this mean?
  - Posix semantics missing for byte range locking and for some unlink and rename cases
  - Case sensitive opens and create
  - Returning a few fields that current query info levels don't give you
- Actual extensions are small. New “POSIX Contexts” proposed
  - On negprot 'negotiate context' returns whether server supports posix features and which ones so client knows before it does an unrecoverable operation (delete on close etc.) that might act differently than expected due to case sensitivity
  - On open/create 'posix create context' is sent
- Discussions continue … very exciting possibilities

Testing … Testing … Testing ...

- Xfstest is VERY helpful
- We are seeing benefits of more automated reconnection testing, and more focus on SMB3 testing
- More automation will continue to help
- What would you like to see?
  - What do you think would help improve testing even more?

- The Future of SMB3 and Linux is very bright
- Let's continue its improvement!

Thank you for your time



Additional Resources to Explore for SMB3 and Linux

-
  - https://msdn.microsoft.com/en-us/library/gg685446.aspx
    - In particular MS-SMB2.pdf at https://msdn.microsoft.com/en-us/library/cc246482.aspx
  - http://www.samba.org
  - Linux CIFS client https://wiki.samba.org/index.php/LinuxCIFS
  - Samba-technical mailing list and IRC channel
  - And various presentations at http://www.sambaxp.org and Microsoft channel 9 and of course SNIA … http://www.snia.org/events/storage-developer
  - And the code:
    - https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/fs/cifs
    - For pending changes, soon to go into upstream kernel see:
      - https://git.samba.org/?p=sfrench/cifs-2.6.git;a=shortlog;h=refs/heads/for-next