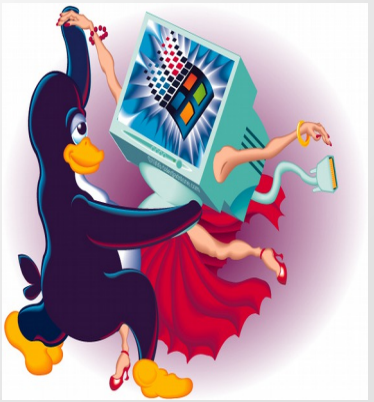
Accessing Samba from Linux. What's new? What's faster? What's better?

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

Most Active Linux Filesystems this year

- 4412 kernel filesystem changesets in last year (since 4.0 kernel)!
  - Linux kernel file system activity is continuing to be strong
  - 5.3% of overall kernel changes (which are dominated by drivers) but watched carefully
  - Iimprovements in defacto standard Linux xfstest test suite as well
- cifs.ko (cifs/smb3 client) had fewer changes than last year but still among more active fs
  - Btrfs 764 changesets (increased)
  - VFS (overall fs mapping layer and common functions) 709 (increased)
  - Xfs 395 (decreased)
  - Nfs client 433
  - Ext4 304 (increased)
  - CIFS/SMB2/SMB3 client 108 (decreased)
  - Nfs server 142 (decreased)
- NB: Samba (cifs/smb2/smb3 server) is more active than all those put together since it is broader in scope (by a lot) and also is in user space not in kernel

Kernel (including cifs client) improving

- 13 months ago we had Linux 4.1 ie “Hurr Durr I'm a Sheep”

Now we have 4.6-rc7 “Charred Weasel”

High Level View of SMB3 Status

- SMB3 support is solid (and large file I/O FAST!), but lacks some optional advanced features (witness protocol integration e.g.) and a few basic features (ACL integration)
  - Metadata performance expected to be slower (need to add open/query compounding)
- SMB3 faster than CIFS (and sometimes NFS) for large file I/O
- SMB3 posix emulation is ok (use mount options “sfu” and “mfsymlinks”) but worse the cifs to Samba (and nfs)
- Can mount with SMB2.02, SMB2.1, SMB3, SMB3.02, 3.1.1
  - Specify vers=2.0 or vers=2.1 or 3.0 or 3.02 or 3.1.1 on mount

Improvements by release

- 3.19 26 changesets
  - Fix Oplock bug, inode caching bug and ioctl clone bug
  - Fix conflicts between SecurityFlags (which allowed CONFIG_MUST_LANMAN and CONFIG_MUST_PLNTXT
  - Improve fallocate support
- Linux 4.0 21 changesets
  - Various minor stability fixes
- Linux 4.1 (23 changesets)
  - Stability fixes: Mapchars fix, fix to allow Unicode surrogate pairs (improved character conversion for some Asian languages), DFS fix, inode number reuse fix
- Linux 4.2 (14)
  - SMB 3.11 (Windows 10) dialect support (improved security)
  - Faster copy offload (REFLINK, duplicate_extents) added for Windows Server 2016

Improvements by release (continued)

- 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 create context (durable v2)
- 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
  - Persistent handle reconnect fixes and improved Mac POSIX support (expected)

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

CopyChunk server (to NTFS) – times vary less new vs. existing target

```
root@ubuntu:~/xfstests-new/xfstests-dev/src# time dd if=/dev/zero of=/mnt1/200M
count=100 bs=2M
100+0 records in
100+0 records out
209715200 bytes (210 MB) copied, 5.3544 s, 39.2 MB/s

real    0m5.363s
user    0m0.000s
sys     0m4.643s
root@ubuntu:~/xfstests-new/xfstests-dev/src# mount -t cifs //192.168.93.142/publ
ic /mnt1 -o username=Administrator,vers=3.02,noperm,sfu^C
root@ubuntu:~/xfstests-new/xfstests-dev/src# ^C
root@ubuntu:~/xfstests-new/xfstests-dev/src# time ./cloner /mnt1/200M /mnt1/copy
chunk-of-200M

real    0m0.313s
user    0m0.000s
sys     0m0.032s
root@ubuntu:~/xfstests-new/xfstests-dev/src# time ./cloner /mnt1/200M /mnt1/copy
chunk-of-200M


real    0m0.250s
user    0m0.000s
sys     0m0.028s
root@ubuntu:~/xfstests-new/xfstests-dev/src#
root@ubuntu:~/xfstests-new/xfstests-dev/src# time ./cloner /mnt1/200M /mnt1/copy
chunk-of-200M-two

real    0m0.335s
user    0m0.000s
sys     0m0.029s
root@ubuntu:~/xfstests-new/xfstests-dev/src# time ./cloner /mnt1/200M /mnt1/copy
chunk-of-200M-two

real    0m0.240s
user    0m0.000s
sys     0m0.029s
```

Better HA: Persistent and Resilient Handles

- New mount options (and code to add corresponding create contexts etc.)
  - “resilienthandles”
  - “persistenthandles”
- Two needed changes
  - Add channel sequence number on reconnect
  - Improve server to server failover
    - Alternate DFS targets in DFS referrals
    - Witness protocol server or share redirection

fallocate

- 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

Cifs-utils

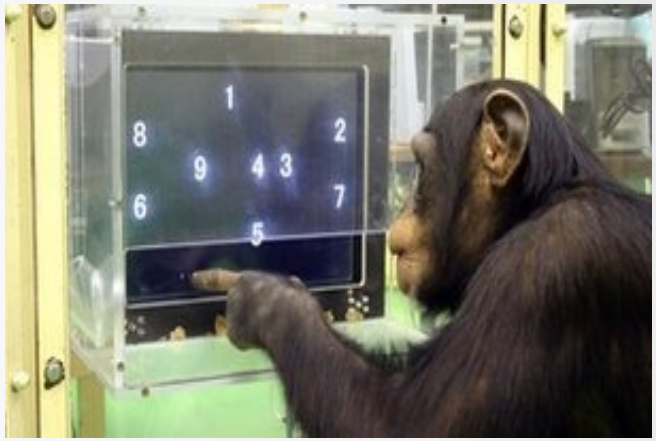
- The userspace utils: mount.cifs, cifs.upcall,set/getcifsacl,cifscreds, idmapwb (idmap plugin),pam_cifscreds
  - thanks to Jeff Layton for maintaining cifs-utils
- 4 changesets over the past year
  - Current version is 6.5
  - Minor bugfixes

Work in Progress

- Xstat integration
  - Returns birth time and dos attributes in more standardized fashion (cifs has a private xattr for that, but few tools use it)
- RichACL integration
- IOCTL to list alternate data streams
  - Querying data in alternate data streams (e.g. for backup) requires disabling posix pathnames (due to conflict with ":")
- Finish up of persistent handle support (adding channel sequence number on reconnect)
- Finish up of encryption support
- Add workaround for guest login problem introduced by "Badlock" Samba security fixes
- DFS improvements, including for DFS reconnect

SMB2/SMB3 Optional Feature Status

- Security
  - Complete: Downgrade attack protection, SMB2.1 signing
  - SMB3.11 negotiate contexts (partial), per-share encryption (started), ACLs (cifs only, started for SMB3)
  - Krb5 and ntlmssp support
  - Not yet: CBAC (DAC ACLs)
- Data Integrity:
  - Durable Handle Support (complete), resilient handles (mount option), persistent handles (need to add channel sequence number on reconnect but mostly complete)
- Performance
  - Complete: multicredit, large I/O
  - Copy offload, and reflink
  - Multichannel (started)
  - Not yet: T10 copy offload, RDMA, directory leases, Branch Cache integration, use of compound ops on wire
- Clustering
  - Not yet: Witness protocol integration
- Other
  - Set/Get Compression and Sparse File support (complete)



POSIX/Linux Compatibility: Details

- *Implemented:*
  - *Hardlinks*
- *Emulated: (current cifs.ko SMB3 code)*
  - *POSIX Path Names:* Approximately 7 reserved characters not allowed in SMB3/NTFS etc. *(e.g. ? * \ : !)*
  - *Symlinks* (ala “mfsymlinks” Minshall-French symlinks, use “mfsymlinks” mount option)
  - *Pseudo-Files:* FIFOs, Pipes, Character Devices (ala “sfu” aka “Microsoft services for unix” use “sfu” mount option)
- *Partial:*
  - *Extended attribute flags* (lsattr/chattr) including compressed flag
  - *POSIX stat and statfs info*
  - *POSIX Byte Range Locks*
- *Not implemented, but emulatable with combination of SMB3 features and/or POSIX Extensions or even use of Apple AAPL create context*
  - *Xattrs* (Security/Trusted for SELinux, User xattrs for apps)
  - *POSIX Mode Bits*
  - *POSIX UID/GID ownership information*
  - *Case Sensitivity* in opening paths
- *Not solvable without additional extensions:*
  - *POSIX Delete (unlink) Behavior*

Approach 1: Enhance support for existing SMB3 features some servers already support

- Get mode from SMB3 ACL (or combination of that and SMB2_CREATE_QUERY_MAXIMAL_ACCESS_REQUEST create context)
- Recognize case sensitive volume at mount time and detect cases where server 'lies' about it
- Cleanup Microsoft “nfs symlink” code to better recognize this symlink (reparse point)
- Implement level 11 SMB2_QUERY_FS_INFO in Samba get “PhsyicalBytesPerSectorForPerformance” and map to statfs f_bsize
- Doesn't address posix byte range locking fully, nor does it always address case sensitive posix path names, nor conflict between streams (which have : separating the file and ADS name) and posix paths (which allow : in the name)

Approach 2

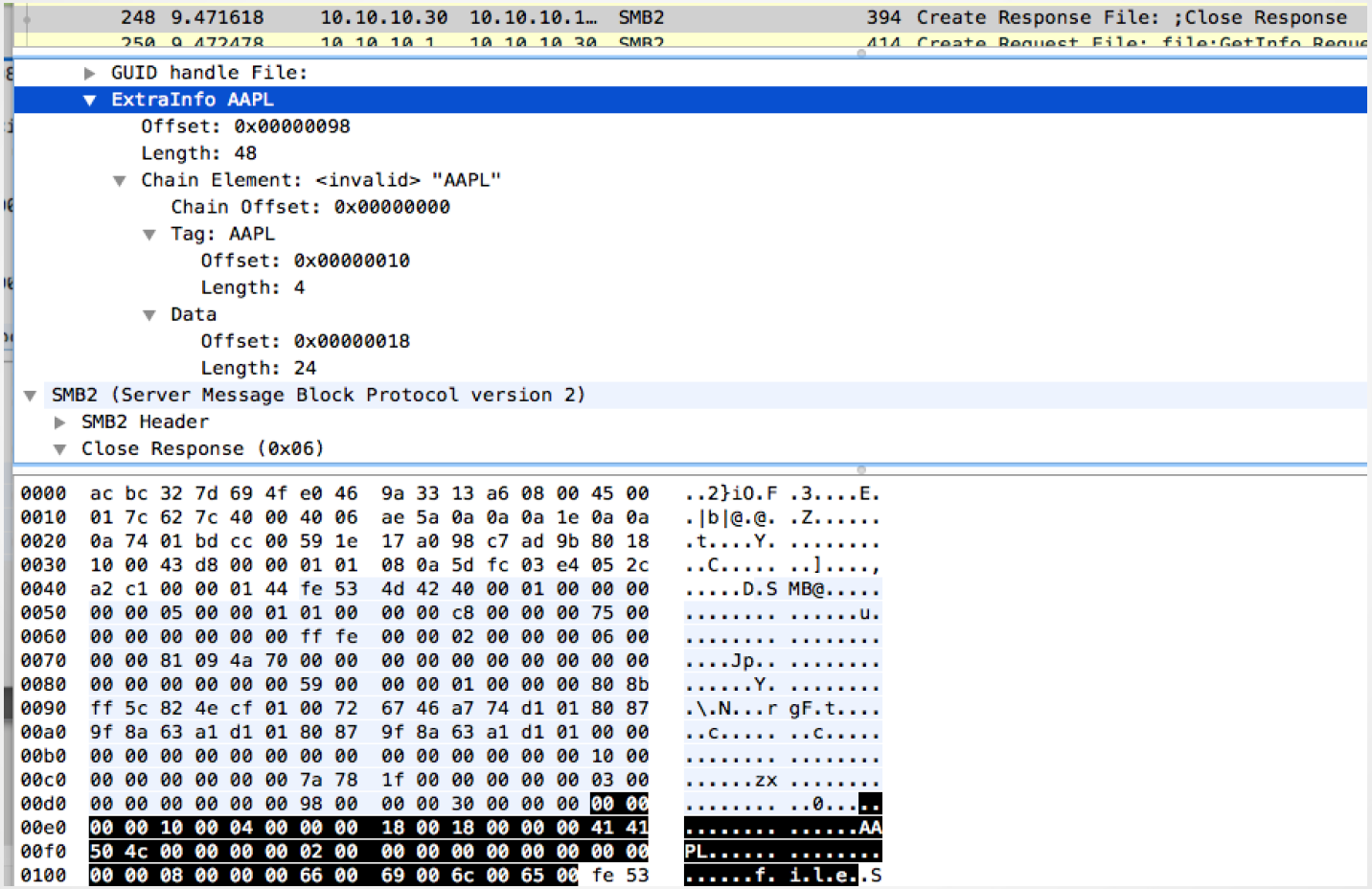
- Implement AAPL context
  - Improved Mac interop is another benefit
  - Samba even has a vfs_fruit module that adds other interesting features (spotlight integration e.g.)
- Subset of POSIX requirements can be solved
- kAAPL_SERVER_CAPS = 0x01,
  - kAAPL_SUPPORTS_READ_DIR_ATTR = 0x01,
  - kAAPL_SUPPORTS_OSX_COPYFILE = 0x02,
  - kAAPL_UNIX_BASED = 0x04
  - kAAPL_SUPPORTS_NFS_ACE = 0x08
- kAAPL_VOLUME_CAPS = 0x02,
  - kAAPL_SUPPORT_RESOLVE_ID = 0x01,
  - kAAPL_CASE_SENSITIVE = 0x02
- kAAPL_MODEL_INFO = 0x04 (pad, length, model string)

Approach 2 (continued) – Mac example

```
fset: 0x00000080
ngth: 40
ain Element: <invalid> "AAPL"
 Chain Offset: 0x00000000
 Tag: AAPL
    Offset: 0x00000010
    Length: 4
 Data
    Offset: 0x00000018
    Length: 16
er Message Block Protocol version 2)
```

```
)a 33 13 a6 ac bc  32 7d 69 4f 08 00 45 00   .F.3.... 2}iO..E.
f1 c2 40 00 40 06  1f 58 0a 0a 0a 74 0a 0a   .8..@.@. .X...t..
:c 00 01 bd 98 c7  ac 97 59 1e 17 a0 80 18   ........ ..Y.....
4e f9 00 00 01 01  08 0a 05 2c a2 c1 5d fc   ..N..... ...,..].
)0 00 01 00 fe 53  4d 42 40 00 01 00 00 00   .......S MB@.....
)5 00 00 01 00 00  00 00 a8 00 00 00 75 00   ........ ......u.
)0 00 00 00 ff fe  00 00 02 00 00 00 06 00   ........ ........
31 09 4a 70 00 00  00 00 00 00 00 00 00 00   ....Jp.. ........
```

Mac example (continued)



Approach 3 – POSIX Extensions for SMB3!

- See Jeremy's talk here and at Vault conference last month

More SMB3 Performance Linux->Linux

- client Ubuntu with 3.16-rc4 with Pavel's patches, srv Fedora 20 (3.14.9 kernel Samba server version 4.1.9)
- dd if=/mnt/testfile of=/dev/null bs=50M count=30
- testfile is 1.5GB existing file, unmount/mount in between each large file copy to avoid any caching effect on client (although server will have cached it)

- SMB3 averaged 199MB/sec reads (copy from server)
- CIFS averaged 170MB/sec reads (copy from server)
- NFSv3 averaged 116MB/sec (copy from server)
- NFSv4 and v4.1 averaged 110MB/sec (copy from server)

- Write speeds (doing dd if=/dev/zero of=/mnt/testfile bs=60M count=25) more varied but averaged similar speeds for copy to server for both NFSv3/v4/v4.1 and SMB3 (~175MB/s)
- NB: Additional NFS server and client scalability patches have recently been added to kernel (it is possible that they may help these cases)

Testing … testing … testing

- Continue work on improving xfstest automation
- Can now use “scratch” mount with cifs.ko expanding the range of xfstests that can run against cifs or smb3 mounts
- Need to cleanup some bugs found by xfstest to remove 'noise' and make it easier to identify and fix any regressions early

XFSTEST details

- Surprising number work even to SMB3 without POSIX support
- Some tests fail due to lack of posix permissions (mode bits) e.g. 29, 30, 67, 84, 87, 88, 98, 109, 123, 126, 129, 317
- Various tests fail due to falloc (missing features, and a bug)
  - 8, 9, 71, 86, 91, 112, 263, 315
- Failures due to other missing posix features
  - Advisory locking (test 131)
- Misc. failures and timestamp coherence client/server
  - Really hard to get mtime consistent on client/server in network file systems
  - 11, 23, 75, 124 ...

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

Thank you for your time