SDC 19
SNIA EMEA
JANUARY 30, 2019
TEL AVIV, ISRAEL
STORAGE DEVELOPER CONFERENCE

# Improved Access to NAS, Windows, Mac and the Cloud from Linux - Review of Recent Progress in SMB3


Steve French
Principal Software Engineer
Azure Storage - Microsoft

# Legal Statement

- This work represents the views of the author(s) and does not necessarily reflect the views of Microsoft 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, Azure and various SMB3/CIFS based NAS appliances)
- Also wrote initial SMB2 kernel client prototype
- Member of Samba team, coauthor of SNIA CIFS Technical Reference,former SNIA CIFS Working Group chair
- Principal Software Engineer, Azure Storage:  Microsoft

# Outline

- General Linux Linux FS and VFS Activity and Status
- What are the goals?
- Key Feature Status
- Features under development, expected soon
- Performance overview
- POSIX compatibility and status of SMB3 Extensions
- Testing

# Outline

- A year ago we had Linux kernel 4.15 “Fearless Coyote”

- Now kernel 5.0-rc4 “Shy Crocodile”

# The ‘real reason’ for kernel 5.0

- Quoting Linus  (January 7th email announcing 5.0-rc1):

“People might well find a feature _they_ like so much that they think it can do as a reason for incrementing the major number. So go wild. Make up your own reason for why it's 5.0."

- Should we claim: “*Version 5.0 marks the reborn, new improved SMB3 Client For Linux*” …?

# What is driving file system activity?

- Proposed new mount and fsinfo API; extending clone API, extending ‘statx’
- Many critical evolving storage features:
  - Better support for faster storage
  - RDMA and low latency ways to access VERY high speed storage (e.g. NVMe), and faster/cheaper (10Gb→40Gb->100Gb) ethernet
  - I/O priority
- Broadening use of copy offload (e.g. fix tools to use “copy_file_range” syscall) and making copy smart
- Cloud: longer latency, object & file coexist, strong security

# Activity since January 2018 (4.15 kernel)

- 5350 kernel file system changes (up 27%) since 4.15 kernel released, 6.2% of kernel overall. FS are important to Linux!
- Kernel is now 17.7 million lines of source code (measured this week with sloccount tool)
- 60+ Linux file systems. cifs.ko (cifs/smb3 client) among more active (#3 in LOC change, #4 in changesets out of 60 and growing). More activity is good!
- BTRFS 1079 changesets (up!), most changesets of any fs related component
- VFS (overall fs mapping layer and common functions) 764, XFS 601 (up), F2FS 423 (up)
- **cifs.ko** (CIFS/SMB2/SMB3 client) 420 changesets (**activity more than doubled**! And continuing to increase)
  - Now 51,609 lines of kernel code (not counting user space helpers and samba tools, kernel similar size to NFS)
- NFS client 285 (down)
- NFS server (including lockd etc.) 125 (down). Linux NFS server **MUCH** smaller than Samba server (or even CIFS or NFS clients).
- And various other file systems: EXT4 222, Ceph 151, GFS2 140, AFS 125 …
- NB: Samba is about as active as all Linux file systems put together - broader in scope (by a lot) and also is user space not kernel. 3.4Million Lines of Code. **100x larger than the NFS server in Linux!**

# Linux File Systems: talented developers

At Linux FS Summit in Utah in April

# Samba team: Amazing group

Some at SMB3 I/O lab in Redmond last fall ...

# What are our goals?

- Make SMB3/SMB3.11 and followons fastest, most secure general purpose way to access file data, whether in cloud, on premises or virtualized
- Implement all reasonable Linux/POSIX features - so apps don't have to know they are running on SMB3 mounts (vs. local)
- As Linux evolves, and need for new features discovered, quickly add support (safely) to kernel client and Samba

# Fixes and Features in progress last year ...

- Lots of completed work!

- ~~Full SMB3.11 support!~~
- ~~Statx (extended stat linux API returning additional metadata flags)~~
- ~~Improved performance~~
- ~~RDMA (smbdirect)~~
- ~~Improved POSIX compatibility~~ (see talk yesterday)
- ~~security improvements~~
- ~~Multidialect support~~
- ~~snapshots~~

# Exciting Year!

- Faster performance
- POSIX Extensions (finally)!
- SMB3.11, improved security
- LOTS of new features

# Quality Much Improved – Top Priority

- More xfstests pass (up to 99 now and growing), vast majority of the rest are skipped due to missing features or being inappropriate for network file systems
- Crediting (flow control) hugely improved (thanks to Pavel Shilovsky and others)
- Many potential issues pointed out by static analysis addressed
- **The "Buildbot"** reduced regressions and is VERY exciting recent addition for CIT (thanks to Ronnie, Aurelien and Paulo)

# 35% more efficient mount & SMB3.11 works!

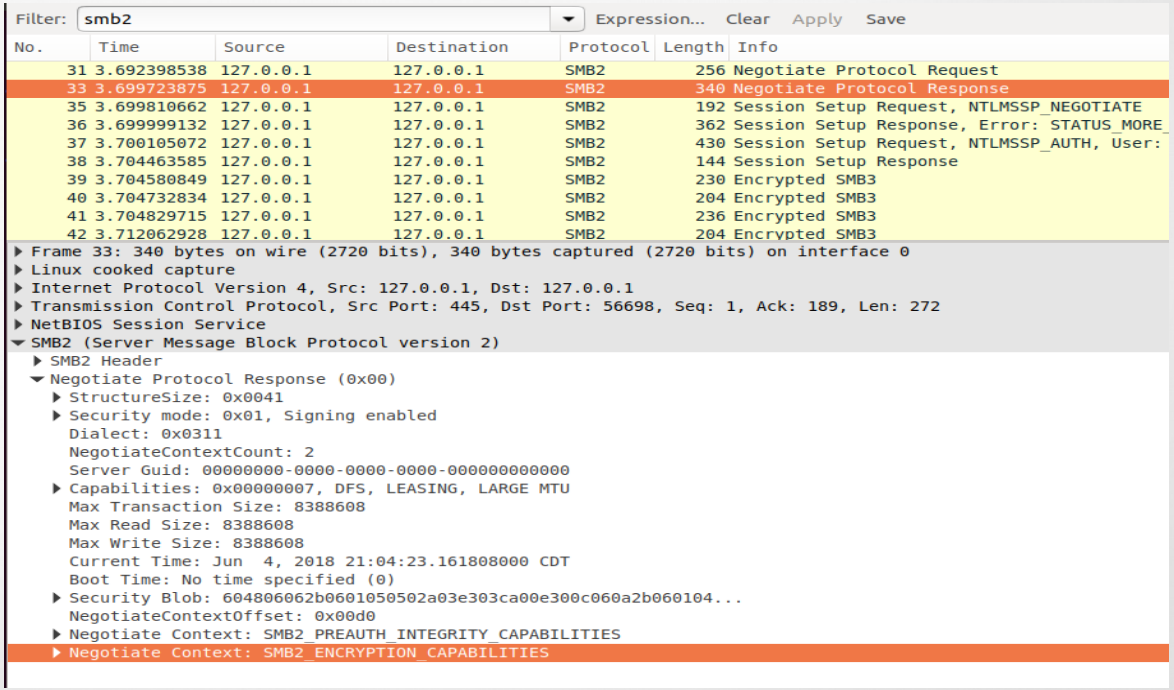
Filter: smb2 Expression... Clear Apply Save

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 4 | 0.000666558 | 172.16.194.1 | 172.16.194.128 | SMB2 | 256 | Negotiate Protocol Request |
| 5 | 0.002358268 | 172.16.194.128 | 172.16.194.1 | SMB2 | 668 | Negotiate Protocol Response |
| 7 | 0.002502467 | 172.16.194.1 | 172.16.194.128 | SMB2 | 192 | Session Setup Request, NTLMSSP_NEGOTIATE |
| 8 | 0.003919218 | 172.16.194.128 | 172.16.194.1 | SMB2 | 382 | Session Setup Response, Error: STATUS_MORE_PROCESSING_REQUIRED, NTL |
| 9 | 0.004131694 | 172.16.194.1 | 172.16.194.128 | SMB2 | 454 | Session Setup Request, NTLMSSP_AUTH, User: \testuser |
| 10 | 0.007151312 | 172.16.194.128 | 172.16.194.1 | SMB2 | 144 | Session Setup Response |
| 11 | 0.007329640 | 172.16.194.1 | 172.16.194.128 | SMB2 | 188 | Tree Connect Request Tree: \\172.16.194.128\IPC$ |
| 12 | 0.007729494 | 172.16.194.128 | 172.16.194.1 | SMB2 | 152 | Tree Connect Response |
| 13 | 0.007898619 | 172.16.194.1 | 172.16.194.128 | SMB2 | 192 | Tree Connect Request Tree: \\172.16.194.128\public |
| 14 | 0.008496801 | 172.16.194.128 | 172.16.194.1 | SMB2 | 152 | Tree Connect Response |
| 15 | 0.008657852 | 172.16.194.1 | 172.16.194.128 | SMB2 | 200 | Create Request File: |
| 16 | 0.009128975 | 172.16.194.128 | 172.16.194.1 | SMB2 | 224 | Create Response File: [unknown] |
| 17 | 0.009318883 | 172.16.194.1 | 172.16.194.128 | SMB2 | 177 | GetInfo Request FS_INFO/FileFsAttributeInformation File: [unknown] |
| 18 | 0.009681622 | 172.16.194.128 | 172.16.194.1 | SMB2 | 164 | GetInfo Response |
| 19 | 0.009836562 | 172.16.194.1 | 172.16.194.128 | SMB2 | 177 | GetInfo Request FS_INFO/FileFsDeviceInformation File: [unknown] |
| 20 | 0.010157145 | 172.16.194.128 | 172.16.194.1 | SMB2 | 152 | GetInfo Response |
| 21 | 0.010309488 | 172.16.194.1 | 172.16.194.128 | SMB2 | 177 | GetInfo Request FS_INFO/FileFsSectorSizeInformation File: [unknown] |
| 22 | 0.010566781 | 172.16.194.128 | 172.16.194.1 | SMB2 | 172 | GetInfo Response |
| 23 | 0.010721458 | 172.16.194.1 | 172.16.194.128 | SMB2 | 240 | Ioctl Request FSCTL_DFS_GET_REFERRALS, File: \172.16.194.128\public |
| 24 | 0.010960930 | 172.16.194.128 | 172.16.194.1 | SMB2 | 145 | Ioctl Response, Error: STATUS_FS_DRIVER_REQUIRED |
| 25 | 0.011248845 | 172.16.194.1 | 172.16.194.128 | SMB2 | 176 | GetInfo Request FILE_INFO/SMB2_FILE_ALL_INFO File: [unknown] |
| 26 | 0.011595834 | 172.16.194.128 | 172.16.194.1 | SMB2 | 248 | GetInfo Response |

```
▶ Frame 5: 668 bytes on wire (5344 bits), 668 bytes captured (5344 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 172.16.194.128, Dst: 172.16.194.1
▶ Transmission Control Protocol, Src Port: 445, Dst Port: 51128, Seq: 1, Ack: 189, Len: 600
▶ NetBIOS Session Service
▼ SMB2 (Server Message Block Protocol version 2)
  ▶ SMB2 Header
  ▼ Negotiate Protocol Response (0x00)
    ▶ StructureSize: 0x0041
    ▶ Security mode: 0x01, Signing enabled
      Dialect: 0x0311
      NegotiateContextCount: 2
      Server Guid: e21779a0-c688-457d-86e9-dd2977809277
    ▶ Capabilities: 0x00000007, DFS, LEASING, LARGE MTU
      Max Transaction Size: 8388608
```

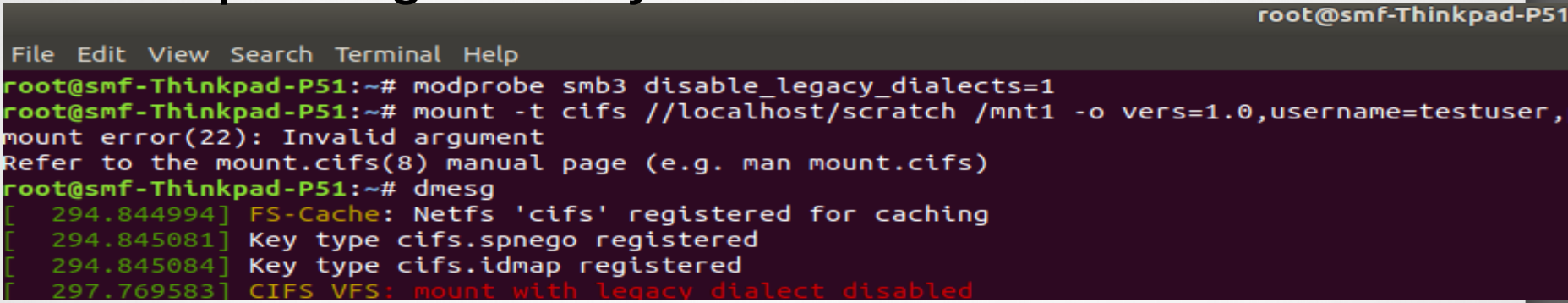
# SMB3.11 AES-CCM encryption works ...

- “mount -t cifs //server/share /mnt -o vers=3.11,seal”
- Thanks Pavel! (and Thank you Aurelien for SMB3.1.1 Auth support)



# Can load it as ‘smb3’ and even disable cifs

- Improving security: can disable cifs



# Current List of CIFS/SMB3 tracepoints and an example of detail for one

```
File  Edit  View  Search  Terminal  Help
root@smf-Thinkpad-P51:/sys/kernel/debug/tracing/events/cifs# ls
enable          smb3_exit_err                 smb3_posix_mkdir_done              smb3_ses_expired
filter          smb3_flush_err                smb3_posix_mkdir_err               smb3_set_info_err
smb3_close_err  smb3_fsctl_err                smb3_query_info_err                smb3_slow_rsp
smb3_cmd_done   smb3_lock_err                 smb3_read_done                     smb3_write_done
smb3_cmd_err    smb3_open_done                smb3_read_err                      smb3_write_err
smb3_enter      smb3_open_err                 smb3_reconnect
smb3_exit_done  smb3_partial_send_reconnect   smb3_reconnect_with_invalid_credits
root@smf-Thinkpad-P51:/sys/kernel/debug/tracing/events/cifs# ls smb3_cmd_err ; cat smb3_cmd_err/format

enable  filter  format  hist  id  trigger
name: smb3_cmd_err
ID: 2049
format:
        field:unsigned short common_type;       offset:0;       size:2; signed:0;
        field:unsigned char common_flags;       offset:2;       size:1; signed:0;
        field:unsigned char common_preempt_count;       offset:3;       size:1; signed:0;
        field:int common_pid;   offset:4;       size:4; signed:1;

        field:__u32 tid;        offset:8;       size:4; signed:0;
        field:__u64 sesid;      offset:16;      size:8; signed:0;
        field:__u16 cmd;        offset:24;      size:2; signed:0;
        field:__u64 mid;        offset:32;      size:8; signed:0;
        field:__u32 status;     offset:40;      size:4; signed:0;
        field:int rc;   offset:44;      size:4; signed:1;

print fmt: "    sid=0x%llx tid=0x%x cmd=%u mid=%llu status=0x%x rc=%d", REC->sesid, REC->tid, REC->cmd
, REC->mid, REC->status, REC->rc
root@smf-Thinkpad-P51:/sys/kernel/debug/tracing/events/cifs#
```

# Tracing with the new ftrace is so easy ...

```
root@smf-Thinkpac
File  Edit  View  Search  Terminal  Help
root@smf-Thinkpad-P51:~# modprobe smb3
root@smf-Thinkpad-P51:~# trace-cmd start -e cifs
root@smf-Thinkpad-P51:~# mount -t cifs //localhost/test /mnt1 -o username=testuser,passw
root@smf-Thinkpad-P51:~# touch /mnt1/newfile
touch: cannot touch '/mnt1/newfile': Permission denied
root@smf-Thinkpad-P51:~# trace-cmd show
```

# Example output: tracing mount and touch (create file) failure

```
                     | / _---=> hardirq/softirq
                     || / _--=> preempt-depth
                     ||| /     delay
      TASK-PID   CPU#  ||||    TIMESTAMP  FUNCTION
         | |       |   ||||       |         |
 mount.cifs-4557  [005] ....  1370.528512: smb3_enter:     cifs_mount: xid=0
 mount.cifs-4557  [005] ....  1370.528778: smb3_enter:     cifs_get_smb_ses: xid=1
 mount.cifs-4557  [005] ....  1370.536041: smb3_cmd_done:  sid=0x0 tid=0x0 cmd=0 mid=0
 mount.cifs-4557  [005] ....  1370.536324: smb3_cmd_err:   sid=0xfb6289ac tid=0x0 cmd=1 mid=1 status=0xc0000016 rc=-5
 mount.cifs-4557  [005] ....  1370.541155: smb3_cmd_done:  sid=0xfb6289ac tid=0x0 cmd=1 mid=2
 mount.cifs-4557  [005] ....  1370.541181: smb3_exit_done:         cifs_get_smb_ses: xid=1
 mount.cifs-4557  [005] ....  1370.541183: smb3_enter:     cifs_setup_ipc: xid=2
 mount.cifs-4557  [005] ....  1370.541419: smb3_cmd_done:  sid=0xfb6289ac tid=0x92f0b9bb cmd=3 mid=3
 mount.cifs-4557  [005] ....  1370.541588: smb3_cmd_done:  sid=0xfb6289ac tid=0x92f0b9bb cmd=11 mid=4
 mount.cifs-4557  [005] ....  1370.541590: smb3_exit_done:         cifs_setup_ipc: xid=2
 mount.cifs-4557  [005] ....  1370.541591: smb3_enter:     cifs_get_tcon: xid=3
 mount.cifs-4557  [005] ....  1370.541768: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=3 mid=5
 mount.cifs-4557  [005] ....  1370.541873: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=11 mid=6
 mount.cifs-4557  [005] ....  1370.541874: smb3_exit_done:         cifs_get_tcon: xid=3
 mount.cifs-4557  [005] ....  1370.542069: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=5 mid=7
 mount.cifs-4557  [005] ....  1370.542070: smb3_open_done: xid=0 sid=0xfb6289ac tid=0xb02df36d fid=0xf976554e cr_opts=0x0 des_access=0x80
 mount.cifs-4557  [005] ....  1370.542122: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=16 mid=8
 mount.cifs-4557  [005] ....  1370.542140: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=16 mid=9
 mount.cifs-4557  [005] ....  1370.542159: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=16 mid=10
 mount.cifs-4557  [005] ....  1370.542197: smb3_cmd_err:   sid=0xfb6289ac tid=0x92f0b9bb cmd=11 mid=11 status=0xc0000225 rc=-2
 mount.cifs-4557  [005] ....  1370.542198: smb3_fsctl_err: xid=0 sid=0xfb6289ac tid=0x92f0b9bb fid=0xffffffffffffffff class=0 type=0x60194 rc=-2
 mount.cifs-4557  [005] ....  1370.542200: smb3_exit_done:         cifs_mount: xid=0
 mount.cifs-4557  [005] ....  1370.542259: smb3_enter:     cifs_root_iget: xid=4
 mount.cifs-4557  [005] ....  1370.542310: smb3_cmd_done:  sid=0xfb6289ac tid=0xb02df36d cmd=16 mid=12
 mount.cifs-4557  [005] ....  1370.542317: smb3_exit_done:         cifs_root_iget: xid=4
      touch-4562  [001] ....  1377.479938: smb3_enter:     cifs_atomic_open: xid=5
      touch-4562  [001] ....  1377.480702: smb3_cmd_err:   sid=0xfb6289ac tid=0xb02df36d cmd=5 mid=13 status=0xc0000022 rc=-13
      touch-4562  [001] ....  1377.480707: smb3_open_err: xid=5 sid=0xfb6289ac tid=0xb02df36d cr_opts=0x40 des_access=0x40000080 rc=-13
      touch-4562  [001] ....  1377.480711: smb3_exit_err:  cifs_atomic_open: xid=5 rc=-13
```

# Stats much improved for SMB2/SMB3

```
$ cat /proc/fs/cifs/Stats
Resources in use
CIFS Session: 1
Share (unique mount targets): 2
SMB Request/Response Buffer: 1 Pool size: 5
SMB Small Req/Resp Buffer: 1 Pool size: 30
Total Large 10 Small 490 Allocations
Operations (MIDs): 0

0 session 0 share reconnects
Total vfs operations: 67 maximum at one time: 2
4 slow responses from localhost for command 5
1 slow responses from localhost for command 6
1 slow responses from localhost for command 14
1 slow responses from localhost for command 16

1) \\localhost\test
SMBs: 243
Bytes read: 1024000  Bytes written: 104857600
TreeConnects: 1 total 0 failed
TreeDisconnects: 0 total 0 failed
Creates: 40 total 0 failed
Closes: 39 total 0 failed
```

# Statx (and cifs pseudoxattrs) and get/set real xattrs work

```
root@smf-Thinkpad-P51:/mnt1# setfattr file2 -n user.somexattr -v somevalue
root@smf-Thinkpad-P51:/mnt1# getfattr file2 -d
# file: file2
user.somexattr="somevalue"

root@smf-Thinkpad-P51:/mnt1# ~/statx/test-statx file2 2M
statx(file2) = 0
results=fdf
  Size: 0               Blocks: 0          IO Block: 16384   regular file
Device: 00:38           Inode: 13107206    Links: 1
Access: (0755/-rwxr-xr-x)  Uid:     0   Gid:     0
Modify: 2018-06-05 02:39:25.088837500-0500
Change: 2018-06-05 02:39:25.088837500-0500
 Birth: 2018-05-31 18:06:01.644761500-0500
Attributes: 0000000000000000 (........ ........ ........ ........ ........ ........
statx(2M) = 0
results=fdf
  Size: 2097152         Blocks: 4096       IO Block: 16384   regular file
Device: 00:38           Inode: 13107210    Links: 1
Access: (0755/-rwxr-xr-x)  Uid:     0   Gid:     0
Modify: 2018-06-05 02:41:05.058102400-0500
Change: 2018-06-05 02:41:05.058102400-0500
 Birth: 2018-06-05 02:41:05.054102300-0500
Attributes: 0000000000000000 (........ ........ ........ ........ ........ ........
root@smf-Thinkpad-P51:/mnt1# getfattr 2M -n user.cifs.creationtime -e hex
# file: 2M
user.cifs.creationtime=0xdfff268fa0fcd301

root@smf-Thinkpad-P51:/mnt1# getfattr 2M -n user.cifs.dosattrib -e hex
# file: 2M
user.cifs.dosattrib=0x80000000
```

SDC 19
SNIA EMEA

# SMB3/CIFS Features by kernel release

- 5.0-rc4 (74 changesets)
  - DFS failover support added (can reconnect to alternate DFS target) for higher availability
    - DFS referral caching now possible, cache updated regularly
  - Support for reconnect if server IP address changes (coreq change in user space implemented in latest version of cifs-utils)
  - Performance improvement for get/set xattr (compounding support extended)
  - Many Bug Fixes including critical once for ‘crediting’ (SMB3 flow control) and reducing reconnects, and fixing large file copy in cases where network connection is slow or interrupted, and fix for OFD lock support)

# SMB3/CIFS Features by kernel release

- 4.20 (70 changesets)
  - RDMA and direct i/o performance improvements
  - Much better compounding (create/delete/set/unlink/mkdir/rmdir etc.), huge perf improvements for metadata access
  - Additional dynamic (ftrace) tracepoints
  - Requested rsize/wsize larger (4MB vs. 1MB)
  - Query Info IOCTL passthrough (enables new “smb-info” tool to display useful metadata in much detail and also ACLs etc.)
  - Many Bug Fixes (including for krb5 mounts to Azure)

# SMB3/CIFS Features by kernel release

- 4.19 (69 changesets) (cifs.ko module version 2.13)
  - Snapshot (previous version support)
  - SMB3.1.1 ACL support
  - Compounding for statfs (perf improvement)
  - smb2/smb3 stats and tracepoints much improved
  - Fix statfs output
  - smb3 xattr alias (eg getfattr -n system.smb3_acl /mnt1/file)
  - Allow disable insecure dialect, vers=1.0, in kconfig
  - Bug fixes (signing, firewall, root dir missing file, backup intent, security)

# SMB3/CIFS Features by release (cont)

- 4.16 (68 changesets) – April 1
  - Add splice_write support
  - Add support for smbdirect (SMB3 rdma). Thanks Long Li!
- 4.17 (56 changesets) - June 3 (cifs.ko module version 2.11)
  - Bug fixes
  - Add signing support for smbdirect
  - Add support for SMB3.11 encryption, and preauth integrity
  - SMB3.11 dialect improvements (and no longer marked experimental)
- 4.18 (89 changesets!) - August 12th (cifs.ko module version 2.12)
  - RDMA and Direct I/O improvements (Thank you Long Li!)
  - Bug fixes
  - SMB3 POSIX extensions (initial minimal set, open and neg. context only. Use ‘posix’ mnt parm)
  - Add “smb3” alias to cifs.ko (“insmod smb3” and also allows “mount -t smb3 ...”)
  - Allow disabling less secure dialects through new module install parm (disable_legacy_dialects)
  - Add support for improved tracing (ftrace, trace-cmd) – thanks to XFS developers for good ideas!
  - Cache root file handle, reducing redundant opens, improving perf (Thanks Ronnie!)

# SMB3/CIFS Features: future (5.1 kernel)

- 5.1 Expected in about 14 weeks
  - Support for Windows nfs style symlinks, nfs reparse points (mkfifo/mknod) with smb3 (and later) mounts
  - Alternate way to store mode (Windows/Mac NFS ACE with special SID)
  - Better Kerberos mounts usability

# SMB3/CIFS Linux client bug status

- bugzilla.kernel.org summary
  - 55 bugs open
- bugzilla.samba.org summary
  - 56 bugs open
- Some of these are old and long fixed … Would love help to triage, and close out some of the bugs which have already been addressed.

# New Features!

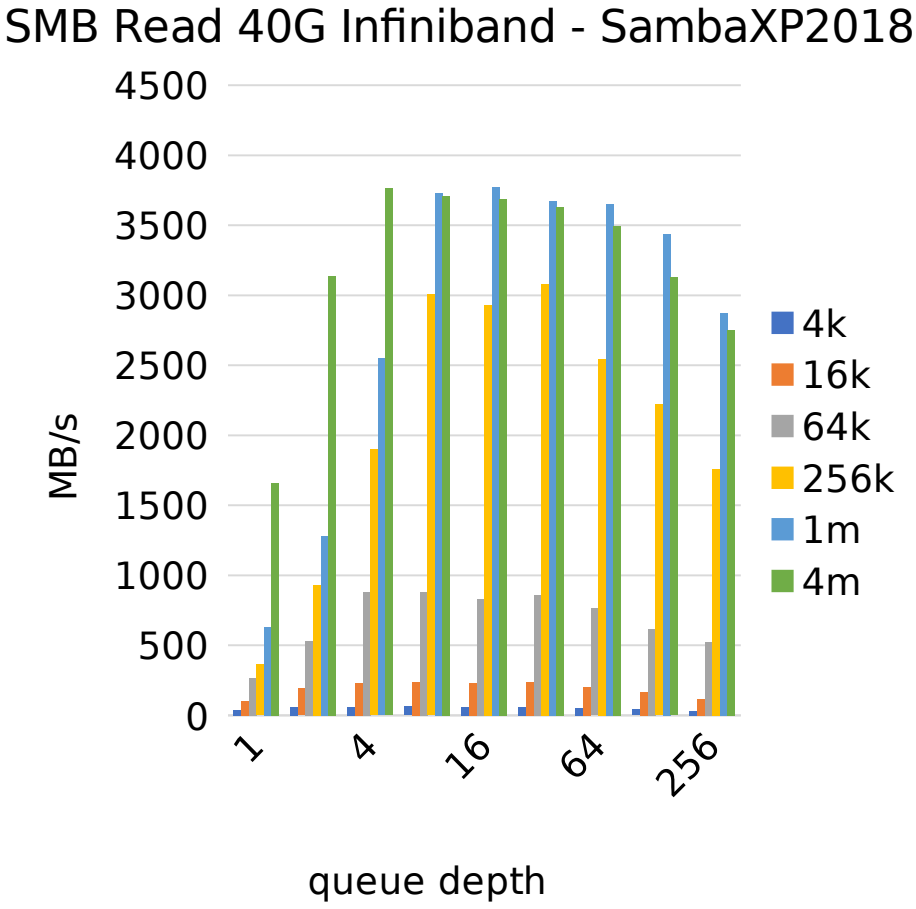
- SMB3 … even better than before!
- smbdirect/RDMA
- Snapshot mounts
- Compounding
- Multichannel
- DFS
- And more ...

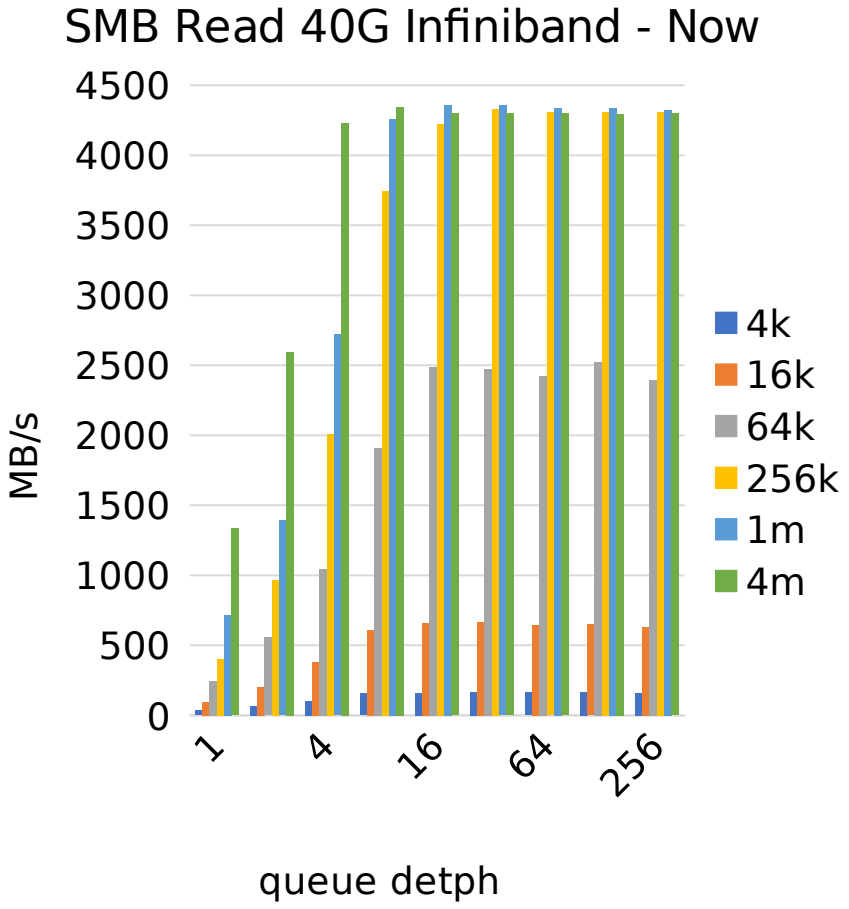
# SMBDIRECT – SMB3 and RDMA

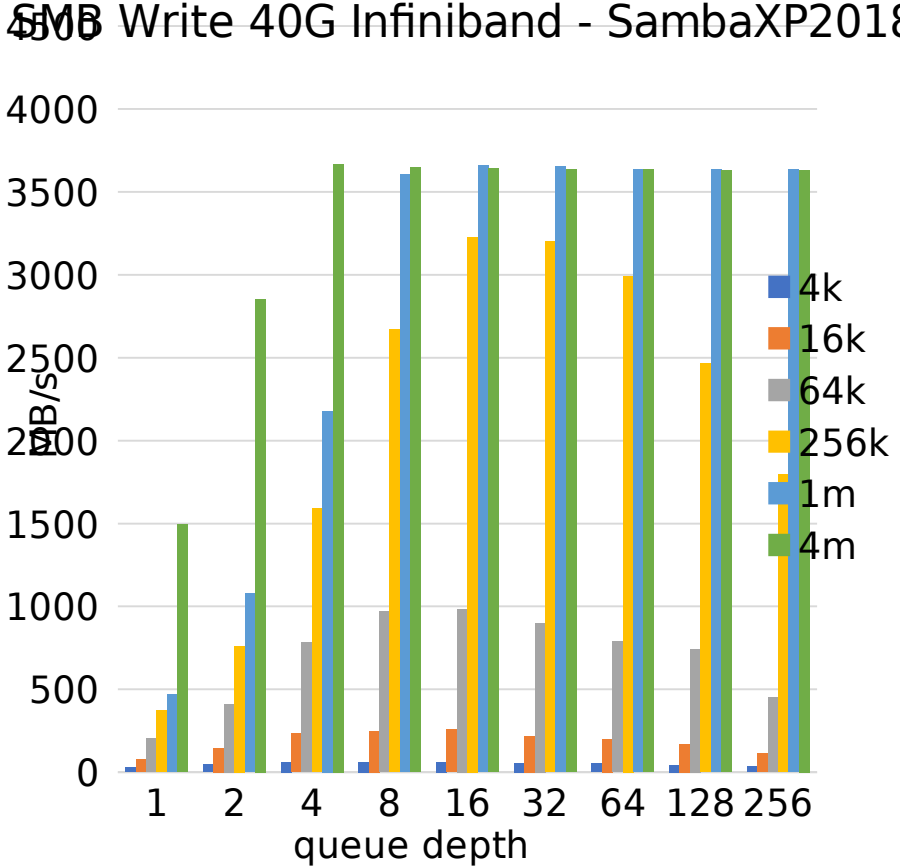
- Thank you Long Li (slides courtesy of him)
- High Speed!

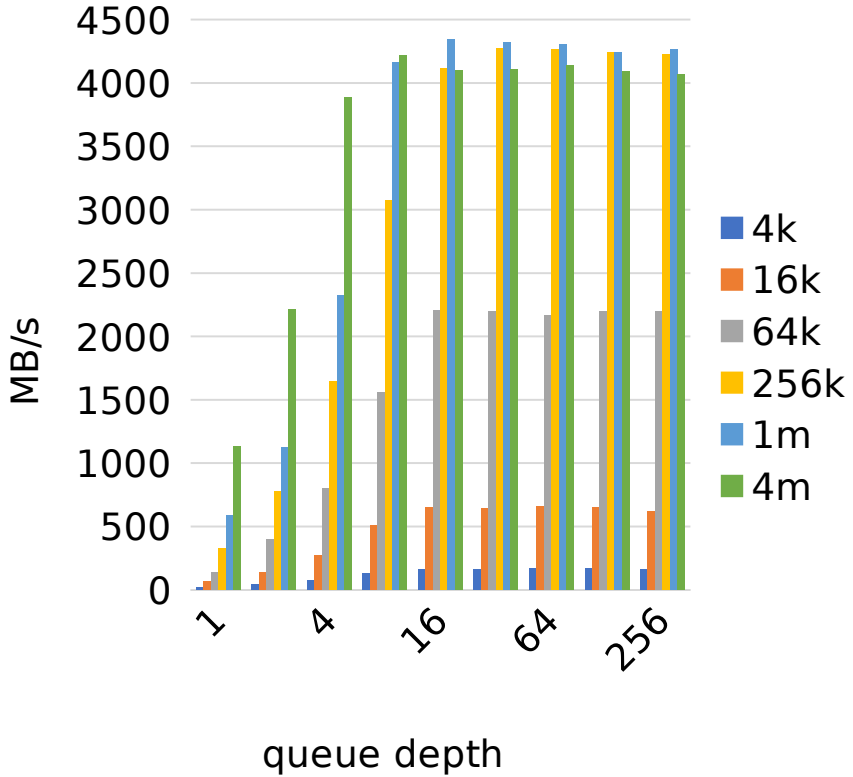
# Test environment

- Hardware
  - Mellanox ConnectX-3 Pro 40G Infiniband
  - Mellanox SX6036 40G VPI switch
  - 2 x Intel E5-2650 v3 @ 2.30GHz
  - 128GB RAM
- Windows 2016 SMB Server
  - SMB Share on RAM disk
- Windows 10 client
  - Registry settings limits to 1 RDMA connection

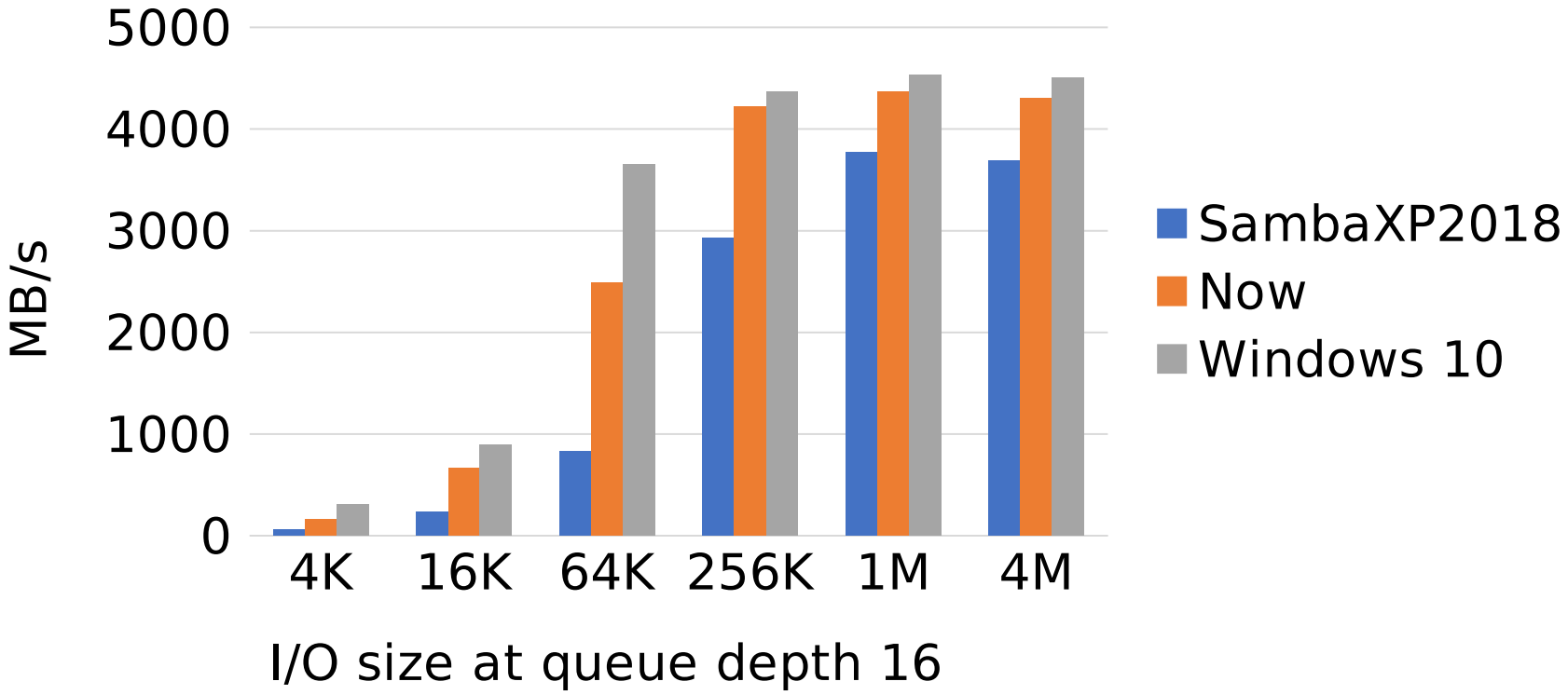

SMB Read 40G Infiniband - SambaXP2018
4500
4000
3500
3000
2500
2000
1500
1000
500
0
MB/s
1
4
16
64
256
queue depth
4k
16k
64k
256k
1m
4m


SMB Read 40G Infiniband - Now
4500
4000
3500
3000
2500
2000
1500
1000
500
0
MB/s
1
4
16
64
256
queue detph
4k
16k
64k
256k
1m
4m


SMB Write 40G Infiniband - SambaXP2018
4500
4000
3500
3000
2500
2000
1500
1000
500
0
MB/s
1
2
4
8
16
32
64
128
256
queue depth
4k
16k
64k
256k
1m
4m


SMB Write 40G Infiniband - Now
4500
4000
3500
3000
2500
2000
1500
1000
500
0
MB/s
1
4
16
64
256
queue depth
4k
16k
64k
256k
1m
4m

# SMB Read 40G Infiniband – comparing to Windows



# Snapshot mounts

- Want to compare backups?
- Look at previous versions?
- Recover corrupted data
- ...
- An example, one mount with “snapshot=” and one without

# Snapshot mounts (example)

```
# cat /proc/mounts | grep cifs
  //172.22.149.186/public /mnt1 cifs ro,vers=default,addr=172.22.149.186,snapshot=131748608570000000,...
  //172.22.149.186/public /mnt2 cifs rw,vers=default,addr=172.22.149.186,...
                root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# ls /mnt1
                  EmptyDir  newerdir
                  root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# ls /mnt1/newerdir
                  root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# ls /mnt2
                  EmptyDir  file  newerdir  newestdir  timestamp-trace.cap
                  root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# ls /mnt2/newerdir
                  new-file-not-in-snapshot
```

# rsize and wsize increase

- Previous default 1MB
  - 4MB gave 1 to 13% improved performance to Samba depending on network speed, 1% better for read.
- Moved to 4MB in 4.20 kernel

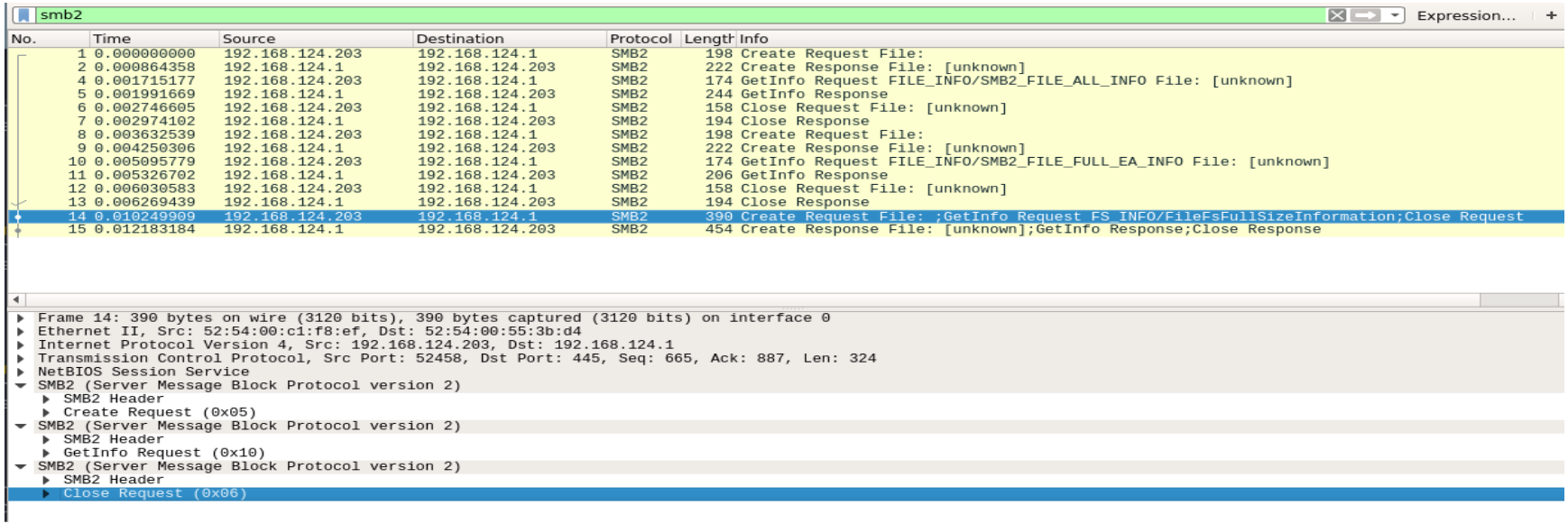
# Compounding – real world scenarios speed up (Thank you Ronnie Sahlberg!)

- Added in so far:
  - update timestamps on existing file: touch /mnt/file"  goes from 6 request/response pairs to 4
  - delete file "rm /mnt/file" from 5 to 2
  - make directory "mkdir /mnt/newdir" 6 to 3
  - remove directory "rmdir /mnt/newdir" 6 down to 2
  - rename goes from 9 request/response pairs to 5 ("mv /mnt/file /mnt/file1")
  - hardlink goes from 8 to only 3 (!)  ("ln /mnt/file1 /mnt/file2")
  - symlink with mfsymlinks enabled goes from 11 to 9 ("ln -s /mnt/file1 /mnt/file3")
  - query file information “stat /mnt/file” goes from six roundtrips down to 2
  - And get/set xattr, and statfs and more

# Compounding

- Many real world scenarios much faster. First two simple examples we tried both more than 1/3 faster
  - Xfstest 013 goes from 171 to 115
  - Xfstest 070 goes from 87 seconds to 47 seconds
  - Note that this is also significantly faster than NFS was (156 seconds) to the same server from the same client

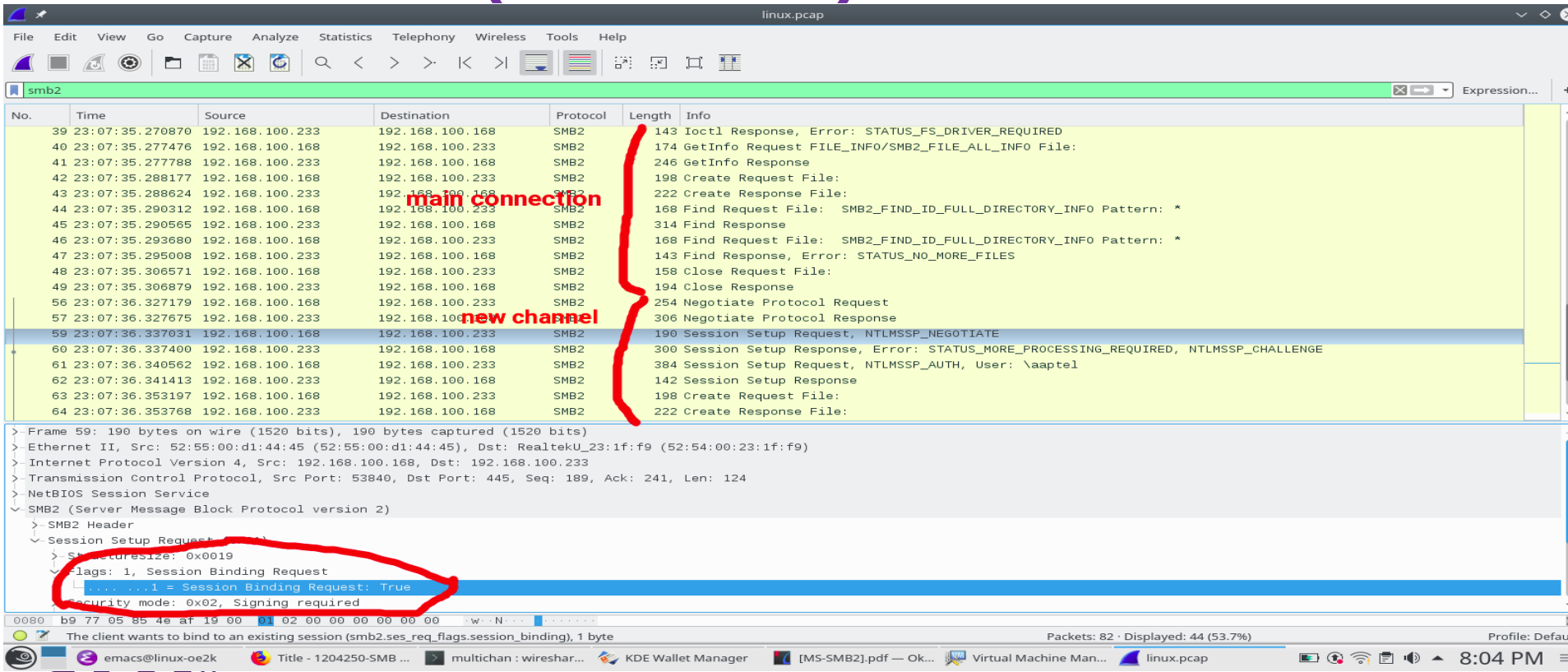
# A compounding example: “df”



# Multichannel

- Thank you Aurelien!
- Made a lot of progress at the Samba test event
  - Server side improvements also in progress (by Metze et al)
- See example wireshark trace showing, 2nd connection opened successfully and used by Linux client (to Windows 2016)

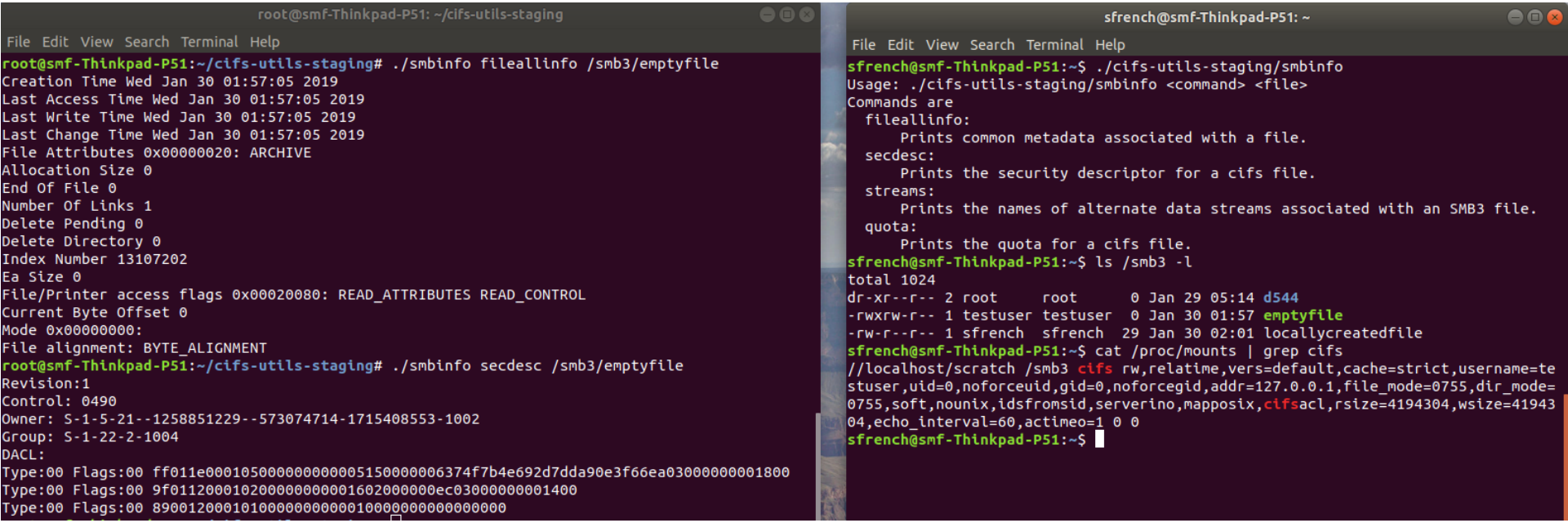
# Multichannel (continued)



# SMB3 Security Features

- SMB3.11 is no longer experimental, is negotiated by default if the server supports it and works well
- SMB3.1.1 secure negotiate works (better than validate negotiate ioctl from SMB2.1 and SMB3)
- SMB3 and SMB3.11 Share Encryption works
  - AES128-CCM encryption algorithm is negotiated (AES128-GCM not supported yet for Linux client or Samba)
- And we made it even easier to disable cifs (vers=1.0)!

# smbinfo: new helper utility for SMB3 mounts



Existing utilities like getcifsacl/setcifsacl can be very helpful
And likely will be extended as well

```
# getcifsacl /smb3/file
REVISION:0x1
CONTROL:0x9004
OWNER:SMF-THINKPAD-P51\testuser
GROUP:Unix Group\testuser
ACL:SMF-THINKPAD-P51\testuser:ALLOWED/0x0/0x1e01ff
ACL:Unix Group\testuser:ALLOWED/0x0/RW
ACL:\Everyone:ALLOWED/0x0/R
```

# passthrough ioctl … and new userspace helper

- Passthrough “query info” call (Thank you Ronnie!)
- New “smb-info” tool
- Also Passthrough fsctl call (ioctl → smb3 fsctl) – prototype in progress
- Many interesting, useful features
  - Now we just need more updates to smb-info and more python or C user space helpers

## Other Optional features

- statfs integration and new mount api integration
  - New API in Al Viro's tree
- IOCTLs e.g. to list alternate data streams
  - NB: Querying data in alternate data streams (e.g. for backup) requires disabling posix pathnames (due to conflict with ":")
- Clustering, Witness protocol integration, multichannel
- Performance feaures
- Other suggestions ...

# POSIX Extensions for SMB3!

- See POSIX Extensions talk here!
- But here are some examples of improvements (even with current kernel, without all the extensions checked in)
- Remember that many ‘posix’ features already work even without the extensions
  - POSIX mapping of reserved characters
  - Two flavors of symlinks recognized
    - Client only (‘mfsymlinks’ ala Macs)
    - Server symlinks (Windows symlinks)
  - Hardlinks
  - Case sensitivity can be set on some server’s shares

```
root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# cat /proc/mounts | grep cifs
//localhost/test-no-posix /mnt1 cifs rw,relatime,vers=3.1.1,cache=strict,username=testuser,domain=,uid=0,noforc
euid,gid=0,noforcegid,addr=127.0.0.1,file_mode=0755,dir_mode=0755,soft,nounix,serverino,mapposix,rsize=1048576,
wsize=1048576,echo_interval=60,actimeo=1 0 0
//localhost/test /mnt cifs rw,relatime,vers=3.1.1,cache=strict,username=testuser,domain=,uid=0,noforceuid,gid=0
,noforcegid,addr=127.0.0.1,file_mode=0755,dir_mode=0755,soft,posix,posixpaths,serverino,mapposix,rsize=1048576,
wsize=1048576,echo_interval=60,actimeo=1 0 0
root@Ubuntu-17-Virtual-Machine:~/cifs-2.6# cat /proc/fs/cifs/DebugData
Display Internal CIFS Data Structures for Debugging
---------------------------------------------------
CIFS Version 2.12
Features: dfs fscache lanman posix spnego xattr acl
Active VFS Requests: 0
Servers:
Number of credits: 16 Dialect 0x311 posix
1) Name: 127.0.0.1 Uses: 2 Capability: 0x300047 Session Status: 1       TCP status: 1
        Local Users To Server: 1 SecMode: 0x1 Req On Wire: 0
        Shares:
        0) IPC: \\127.0.0.1\IPC$ Mounts: 1 DevInfo: 0x0 Attributes: 0x0
        PathComponentMax: 0 Status: 1 type: 0
        Share Capabilities: None         Share Flags: 0x0
        tid: 0x4f5511db Maximal Access: 0x1f00a9

        1) \\localhost\test Mounts: 1 DevInfo: 0x20 Attributes: 0x1006f
        PathComponentMax: 255 Status: 1 type: DISK
        Share Capabilities: None Aligned, Partition Aligned,    Share Flags: 0x0
        tid: 0x8579c31d Optimal sector size: 0x200      Maximal Access: 0x1f01ff

        2) \\localhost\test-no-posix Mounts: 1 DevInfo: 0x20 Attributes: 0x1006f
        PathComponentMax: 255 Status: 1 type: DISK
        Share Capabilities: None Aligned, Partition Aligned,    Share Flags: 0x0
        tid: 0x1813a493 Optimal sector size: 0x200      Maximal Access: 0x1f01ff

        MIDs:
```

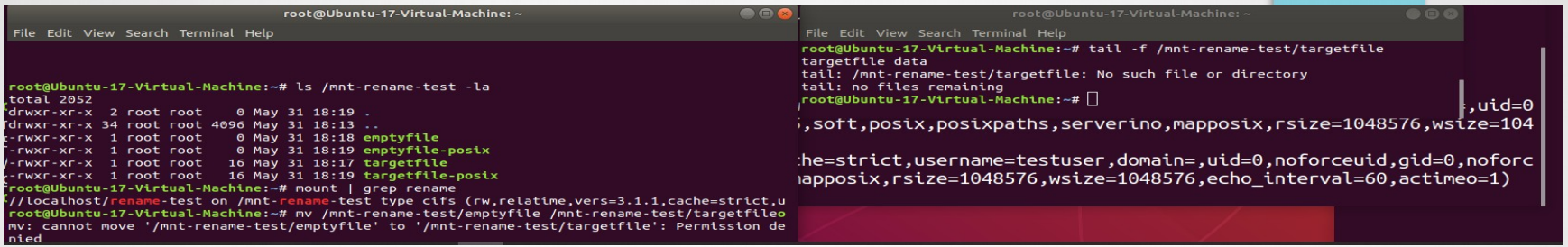
# Mode bits work on create and mkdir

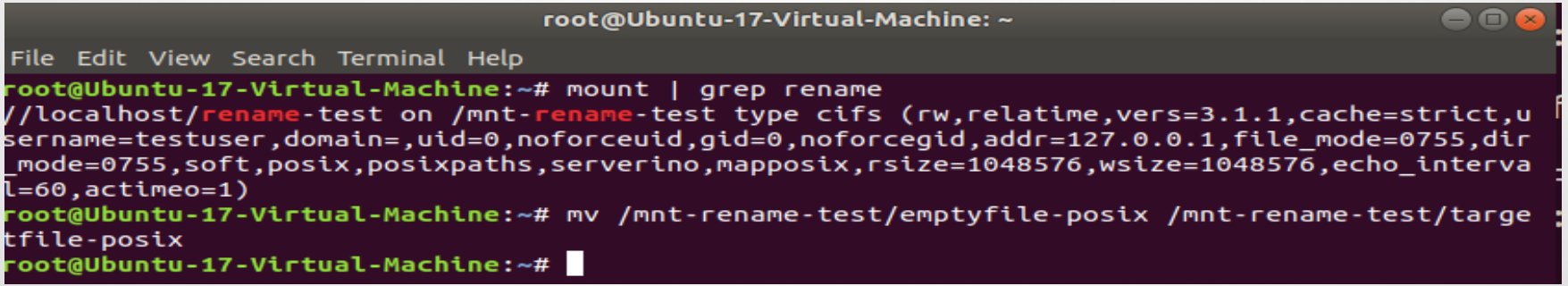
```
root@Ubuntu-17-Virtual-Machine:/mnt# ~/create-4-files-with-mode-test
root@Ubuntu-17-Virtual-Machine:/mnt# cd /mnt1
root@Ubuntu-17-Virtual-Machine:/mnt1# ~/create-4-files-with-mode-test
root@Ubuntu-17-Virtual-Machine:/mnt1# ls /test /test-no-posix -la
/test:
total 12
drwxrwxrwx  3 root     root     4096 May 31 16:55 .
drwxr-xr-x 32 root     root     4096 May 31 16:46 ..
-rwx------  1 testuser testuser    0 May 31 16:55 0700
-rwxrwx---  1 testuser testuser    0 May 31 16:55 0770
-rwxrwxr-x  1 testuser testuser    0 May 31 16:55 0775
drwxr-xr-x  2 sfrench  sfrench  4096 Mar 24 10:34 tmp

/test-no-posix:
total 8
drwxrwxrwx  2 root     root     4096 May 31 16:55 .
drwxr-xr-x 32 root     root     4096 May 31 16:46 ..
-rwxrw-r--  1 testuser testuser    0 May 31 16:55 0700
-rwxrw-r--  1 testuser testuser    0 May 31 16:55 0770
-rwxrw-r--  1 testuser testuser    0 May 31 16:55 0775
root@Ubuntu-17-Virtual-Machine:/mnt1# mkdir UPPER
root@Ubuntu-17-Virtual-Machine:/mnt1# touch upper
root@Ubuntu-17-Virtual-Machine:/mnt1# cd /mnt
root@Ubuntu-17-Virtual-Machine:/mnt# mkdir UPPER
root@Ubuntu-17-Virtual-Machine:/mnt# touch upper
root@Ubuntu-17-Virtual-Machine:/mnt# ls /test /test-no-posix
/test:
0700  0770  0775  tmp  upper  UPPER

/test-no-posix:
0700  0770  0775  UPPER
```

# Rename works with POSIX extensions!



# SMB3 Performance – the Myth

- Googling NFS vs. SMB3 (or Samba) ... first result said:

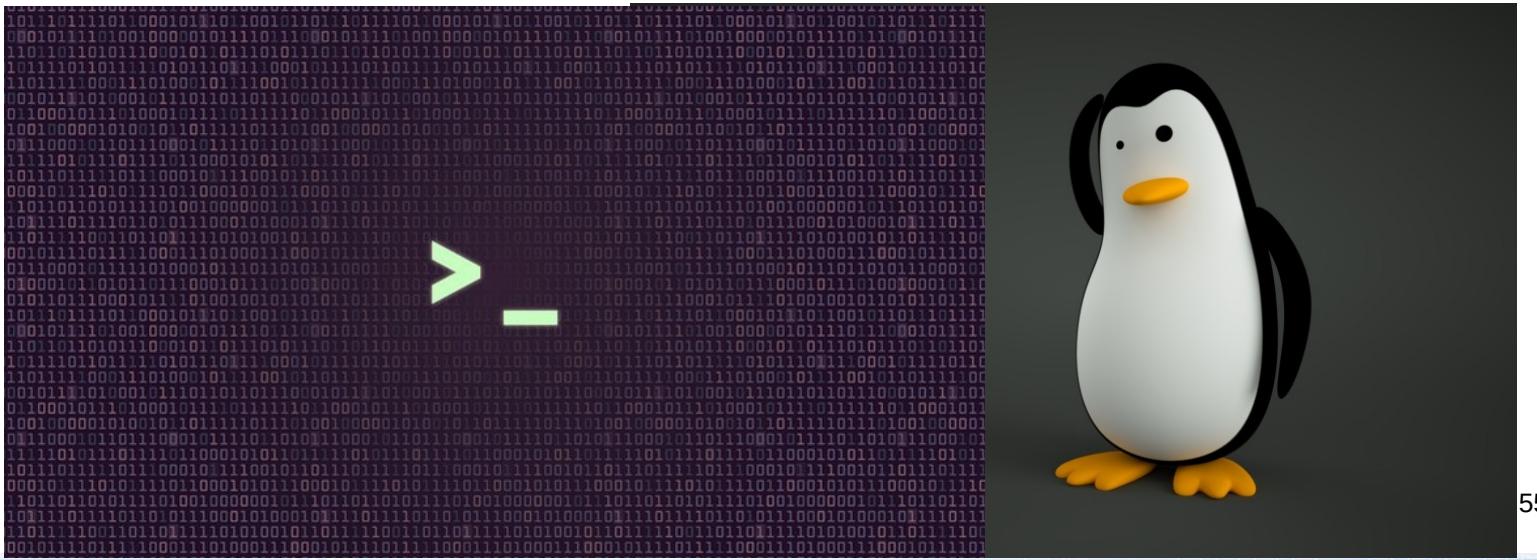
  *"As you can see NFS offers a better performance and is unbeatable if the files are medium sized or small. If the files are large enough the timings of both methods get closer to each other. Linux and Mac OS owners should use NFS instead of SMB. Sadly Windows users are forced to use SMB ..."*


MYTHBUSTERS

# SMB3 Performance

- As described and demonstrated at the last SDC and also at the Redmond event there are various cases where SMB3 is faster than NFS (Linux to Linux!) especially where SMB3 performance features including compounding and larger I/O match the workload well
- Even some common (and simple) copy scenarios can be > 20% faster over SMB3
- And we are improving SMB3 client at a rapid pace!

# Some suggestions on configuration and mount options for optimal use of SMB3 on Linux

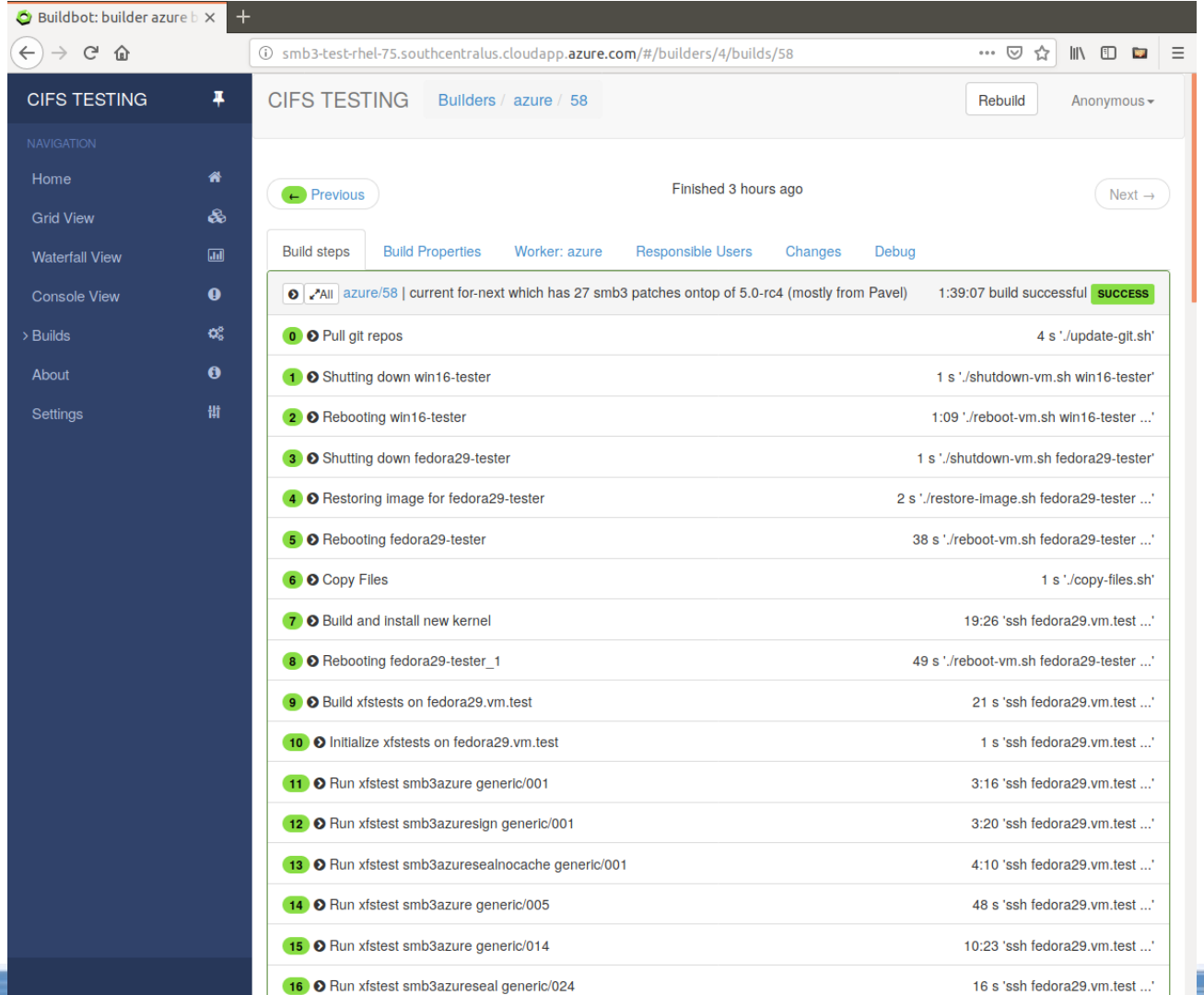


# Still a lot of work to do though! SMB3 Performance WIP: Great Features… but only if we implement them!

- Compounding (at lot went in 4.18 and 4.20 … let's keep going)
- Large file I/O (looks good, let's continue to optimize)
- File Leases
  - Lease upgrades
- Directory Leases  (complete for root directory, to be extended ...)
- Handle caching (under investigation)
- Crediting (very helpful feature)
- I/O priority
- Copy Offload
- Multi-Channel (in progress)
  - And optional RDMA (much improved, will be even better in 4.20)
- Linux specific protocol optimizations possible too ...

# Testing

- See xfstesting page in cifs wiki https://wiki.samba.org/index.php/Xfstesting-cifs
- Easy to setup, exclude file for slow tests or failing ones
- XFSTEST status update
  - Bugzillas
  - Features in progress
- The buildbot!
  - http://http://smb3-test-rhel-75.southcentralus.cloudapp.azure.com


Buildbot: builder azure b
smb3-test-rhel-75.southcentralus.cloudapp.azure.com/#/builders/4/builds/58
CIFS TESTING
NAVIGATION
Home
Grid View
Waterfall View
Console View
Builds
About
Settings
CIFS TESTING
Builders / azure / 58
Rebuild
Anonymous
Previous
Finished 3 hours ago
Next →
Build steps
Build Properties
Worker: azure
Responsible Users
Changes
Debug
All azure/58 | current for-next which has 27 smb3 patches ontop of 5.0-rc4 (mostly from Pavel) 1:39:07 build successful SUCCESS
0 Pull git repos 4 s './update-git.sh'
1 Shutting down win16-tester 1 s './shutdown-vm.sh win16-tester'
2 Rebooting win16-tester 1:09 './reboot-vm.sh win16-tester ...'
3 Shutting down fedora29-tester 1 s './shutdown-vm.sh fedora29-tester'
4 Restoring image for fedora29-tester 2 s './restore-image.sh fedora29-tester ...'
5 Rebooting fedora29-tester 38 s './reboot-vm.sh fedora29-tester ...'
6 Copy Files 1 s './copy-files.sh'
7 Build and install new kernel 19:26 'ssh fedora29.vm.test ...'
8 Rebooting fedora29-tester_1 49 s './reboot-vm.sh fedora29-tester ...'
9 Build xfstests on fedora29.vm.test 21 s 'ssh fedora29.vm.test ...'
10 Initialize xfstests on fedora29.vm.test 1 s 'ssh fedora29.vm.test ...'
11 Run xfstest smb3azure generic/001 3:16 'ssh fedora29.vm.test ...'
12 Run xfstest smb3azuresign generic/001 3:20 'ssh fedora29.vm.test ...'
13 Run xfstest smb3azuresealnocache generic/001 4:10 'ssh fedora29.vm.test ...'
14 Run xfstest smb3azure generic/005 48 s 'ssh fedora29.vm.test ...'
15 Run xfstest smb3azure generic/014 10:23 'ssh fedora29.vm.test ...'
16 Run xfstest smb3azureseal generic/024 16 s 'ssh fedora29.vm.test ...'

# Conclusion … When is SMB3 good?

- When need nice security …
- Workloads where performance with lots of large directories is not an obstacle (pending improvements to leasing and compounding in cifs.ko)
- Workloads which do not depend on case sensitivity (common unfortunately) and do not depend on advisory locking or delete of open files (more rare) … pending POSIX extensions in Samba etc.
- Where you can take advantage of smbdirect (RDMA)
- Where global namespace (DFS) helps
- Where rich features of SMB3 (snapshots, encrypted/compressed files, persistent handles) are helpful …
- And of course … to the cloud (Azure) and Macs and Windows and … not just Samba and NAS


+ SMB 3