SMB3.1.1 to the Cloud and Beyond: Update on Progress in SMB3 Support for Linux

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 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,former SNIA CIFS Working Group chair
- Principal Software Engineer, Azure Storage: Microsoft

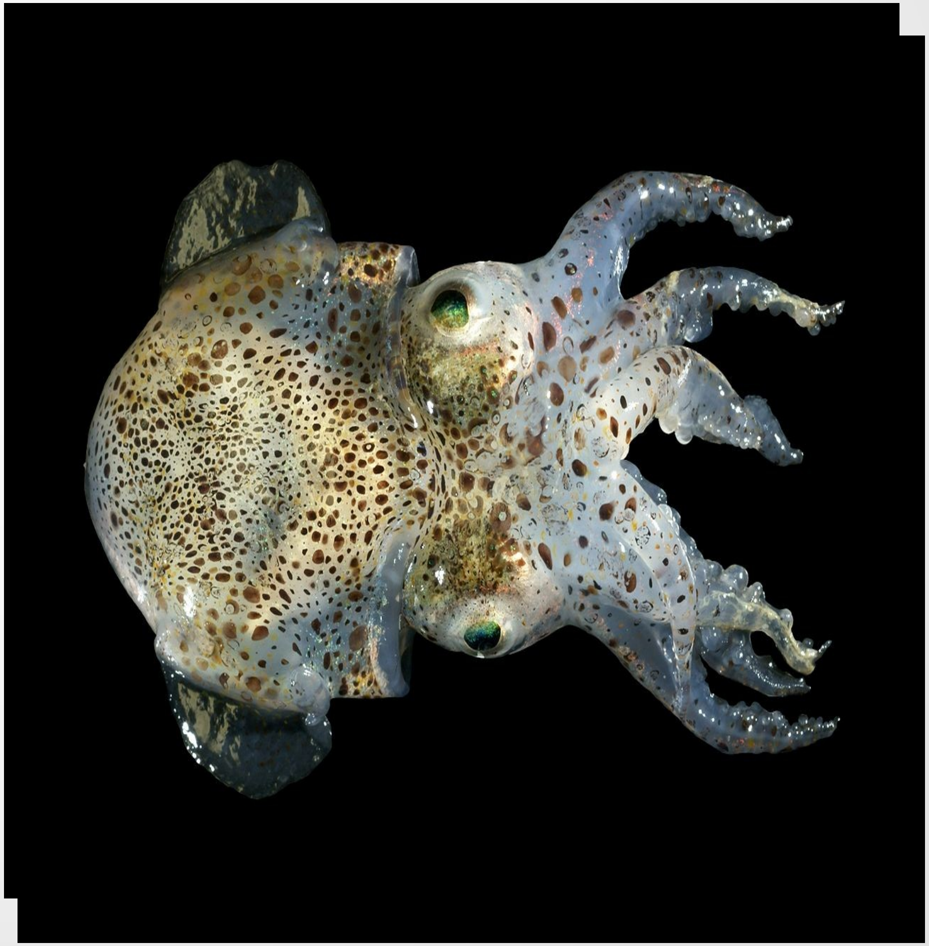
Outline

- General Linux File System Status – Linux FS and VFS Activity
- What are the goals?
- What’s New – Key Feature Status
- Cifs-utils enter a new era …
- Features expected soon …
- Performance Overview
- POSIX Extensions
- Testing Improvements

A year ago … and now … kernel (including SMB3 client cifs.ko) improving

- A year ago Linux 4.18 “Merciless Moray”

Last week:5.3 “Bobtail Squid”



Discussions driving some of the FS development activity ?

- New mount API
- Improving support for Containers
- Better support for faster storage (NVME, RDMA)
- io_uring
- Fixing copy offload (e.g. copy and clone file range syscalls)
- Shift to Cloud (longer latencies, object & file coexisting)

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

2019 Linux FS/MM summit
(last month in Puerto Rico)

- Great group of talented developers

Samba.IO.lab
Welcome

Most Active Linux Filesystems this year

- 5344 kernel filesystem changesets last year (since 4.18 kernel). (down slightly)
  - FS activity: 6% of overall kernel changes (which are dominated by drivers) down slightly as % of activity
  - Kernel is huge (> 18 million lines of code, measured Saturday)
- There are many Linux file systems (>60), but six (and the VFS layer itself) drive two thirds of the activity (btrfs, xfs and cifs are the three most active)
  - File systems represent 5.1% of kernel source code (930KLOC)
- cifs.ko (cifs/smb3 client) among more active fs
  - #3 most active with 466 changesets!
  - 53KLOC (not counting user space cifs-utils which are 10.5KLOC and samba tools which are larger)

Linux File System Change Detail (4.16-now)

- BTRFS 1004 changesets (down)
- VFS (overall fs mapping layer and common functions) 775 (down)
- XFS 521 (up slightly)
- CIFS/SMB2/SMB3 client 466**. And increasing!**
- F2FS 358 (down)
- NFS client 328 (down)
- And others: EXT4 206(down), AFS 155, Ceph 172, GFS2 142, overlayfs 103, ocfs2 108 ...
- NFS server 137 (down). Linux NFS server is **MUCH** smaller than CIFS client or Samba
- NB: Samba is as active as all Linux file systems put together (>5000 changesets per year) - broader in scope (by a lot) and also is user space not kernel. **100x larger than the NFS server in Linux! Samba now 3.5 million lines of code (measured Saturday with sloccount)!**

What are the goals?

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

Examples of Great Progress this year!

Snapshot mounts

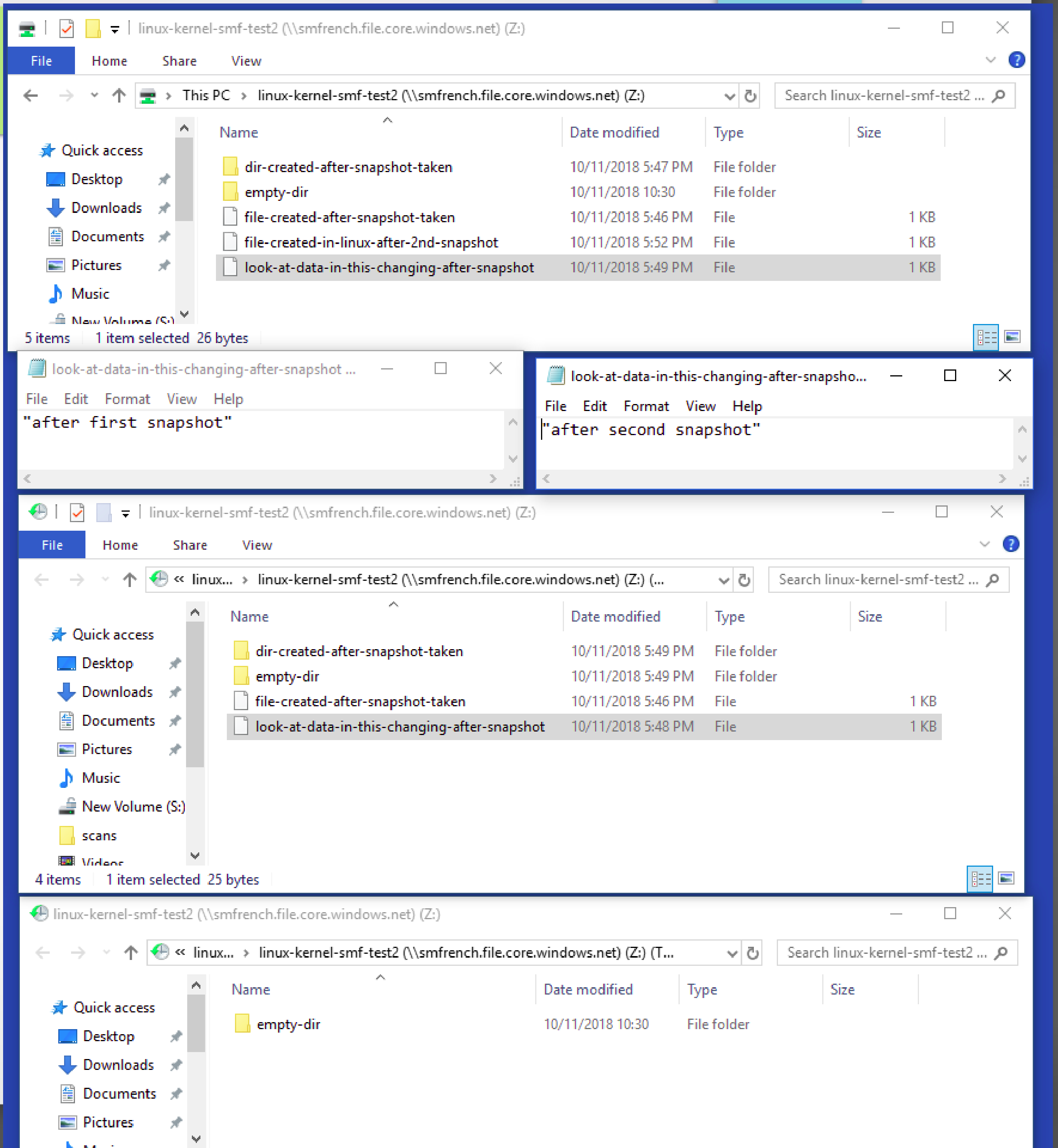
- Want to compare backups?
- Look at previous versions?
- Recover corrupted data
- ...
- An example, one mount with “snapshot=” and one without

```
# 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
```

Snapshots rock!

- Different mounts
- SS is Read-only
- Easy to use



linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:)
File
Home
Share
View
This PC › linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:)
Search linux-kernel-smf-test2 ...
Quick access
Desktop
Downloads
Documents
Pictures
Music
New Volume (S:)

Name	Date modified	Type	Size
dir-created-after-snapshot-taken	10/11/2018 5:47 PM	File folder	
empty-dir	10/11/2018 10:30	File folder	
file-created-after-snapshot-taken	10/11/2018 5:46 PM	File	1 KB
file-created-in-linux-after-2nd-snapshot	10/11/2018 5:52 PM	File	1 KB
look-at-data-in-this-changing-after-snapshot	10/11/2018 5:49 PM	File	1 KB

5 items 1 item selected 26 bytes

look-at-data-in-this-changing-after-snapshot ...
File
Edit
Format
View
Help

"after first snapshot"

"after second snapshot"

linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:)
File
Home
Share
View
« linux... › linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:) (...
Search linux-kernel-smf-test2 ...
Quick access
Desktop
Downloads
Documents
Pictures
Music
New Volume (S:)
scans
Videos

Name	Date modified	Type	Size
dir-created-after-snapshot-taken	10/11/2018 5:49 PM	File folder	
empty-dir	10/11/2018 5:49 PM	File folder	
file-created-after-snapshot-taken	10/11/2018 5:46 PM	File	1 KB
look-at-data-in-this-changing-after-snapshot	10/11/2018 5:48 PM	File	1 KB

4 items 1 item selected 25 bytes

linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:)
« linux... › linux-kernel-smf-test2 (\\smfrench.file.core.windows.net) (Z:) (T...
Search linux-kernel-smf-test2 ...
Quick access
Desktop
Downloads
Documents
Pictures

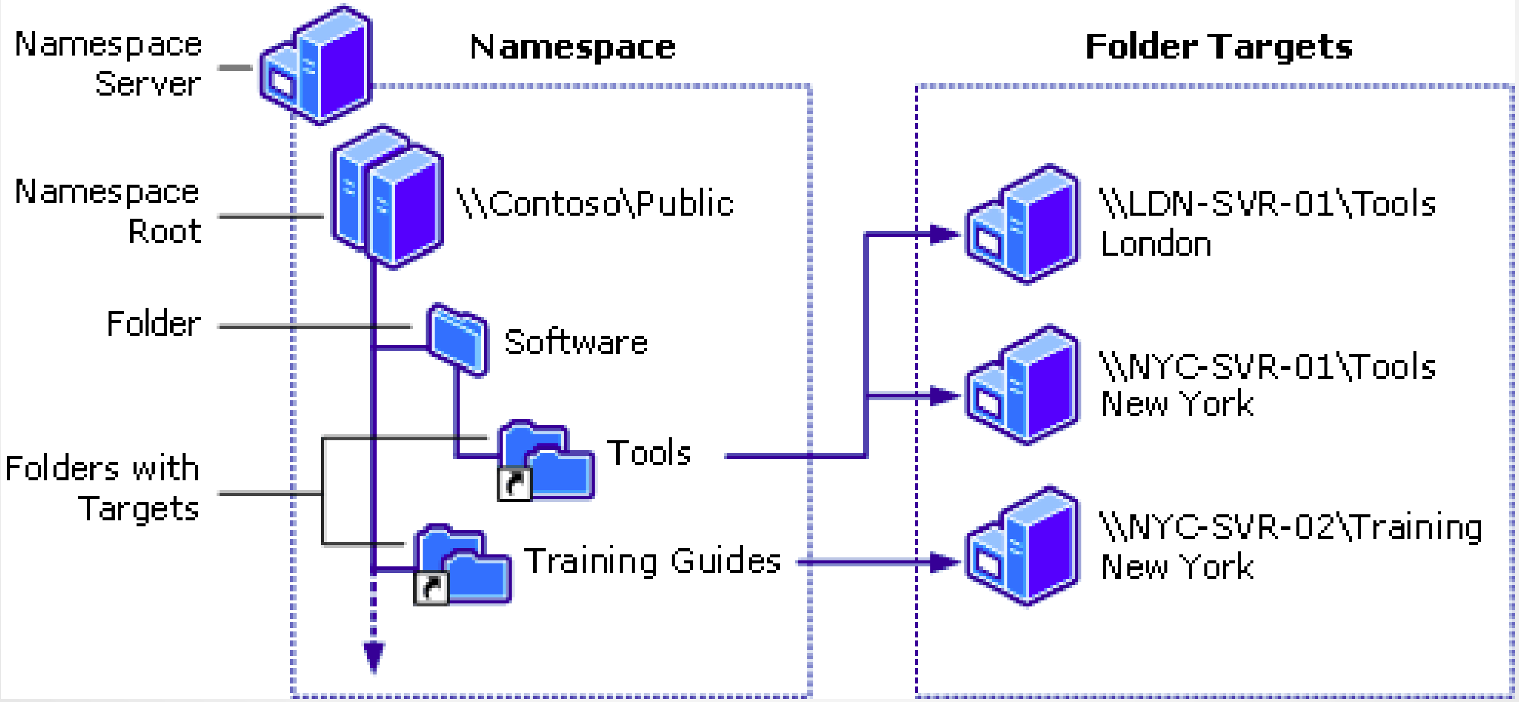
Name	Date modified	Type	Size
empty-dir	10/11/2018 10:30	File folder	

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
- And default block size moved to 1MB (helps cp)

Global Name Space – Much better! Thank you Paulo!

- Remember what DFS could do in Windows … now the Linux client can handle failover and DFS entry caching too



New security models

- Modefromsid
- Multiuser improvements

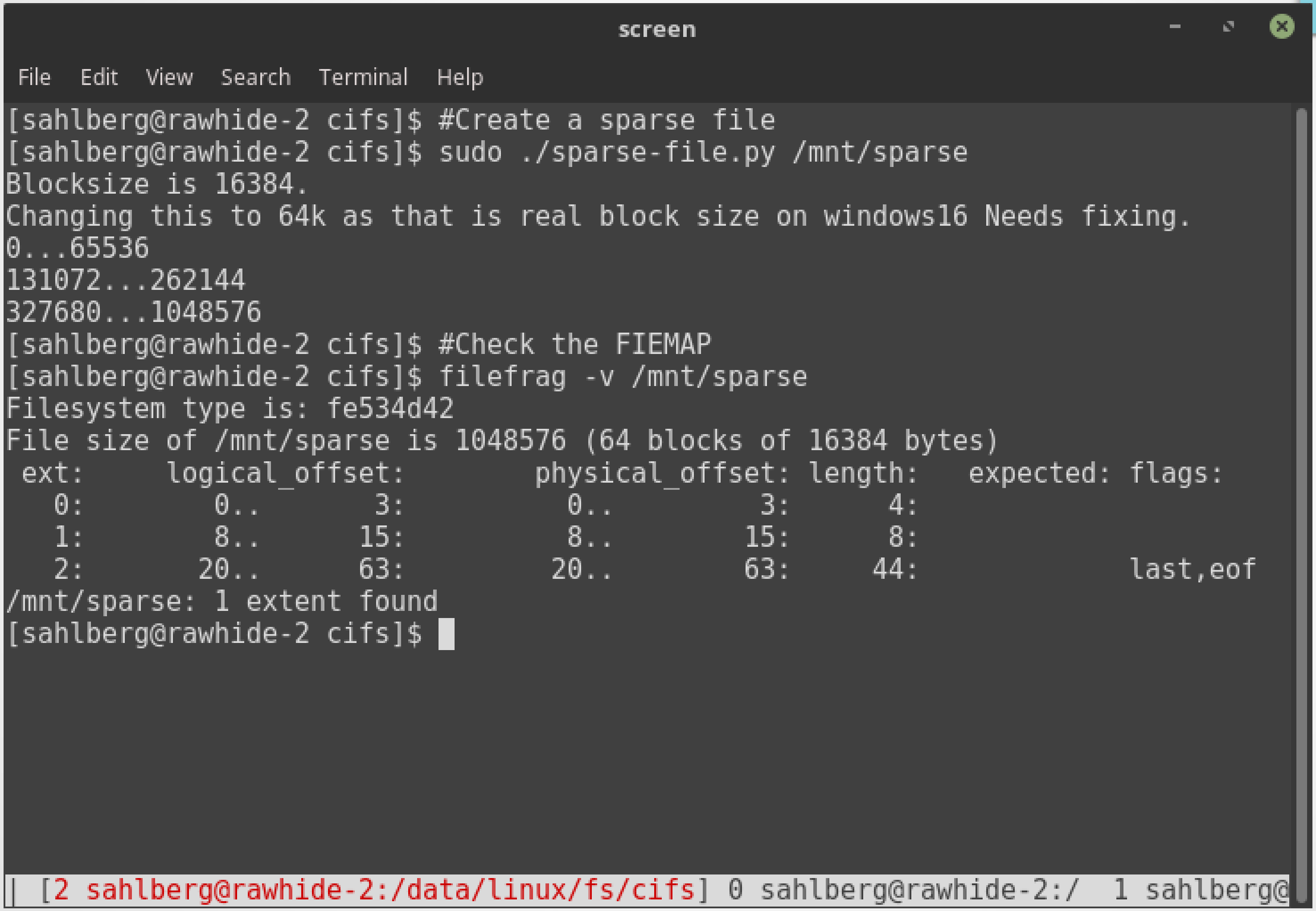
Multichannel

- Thank you Aurelien!
- Expected to be a big performance win …
- Testing continues at SDC SMB3 Plugfest

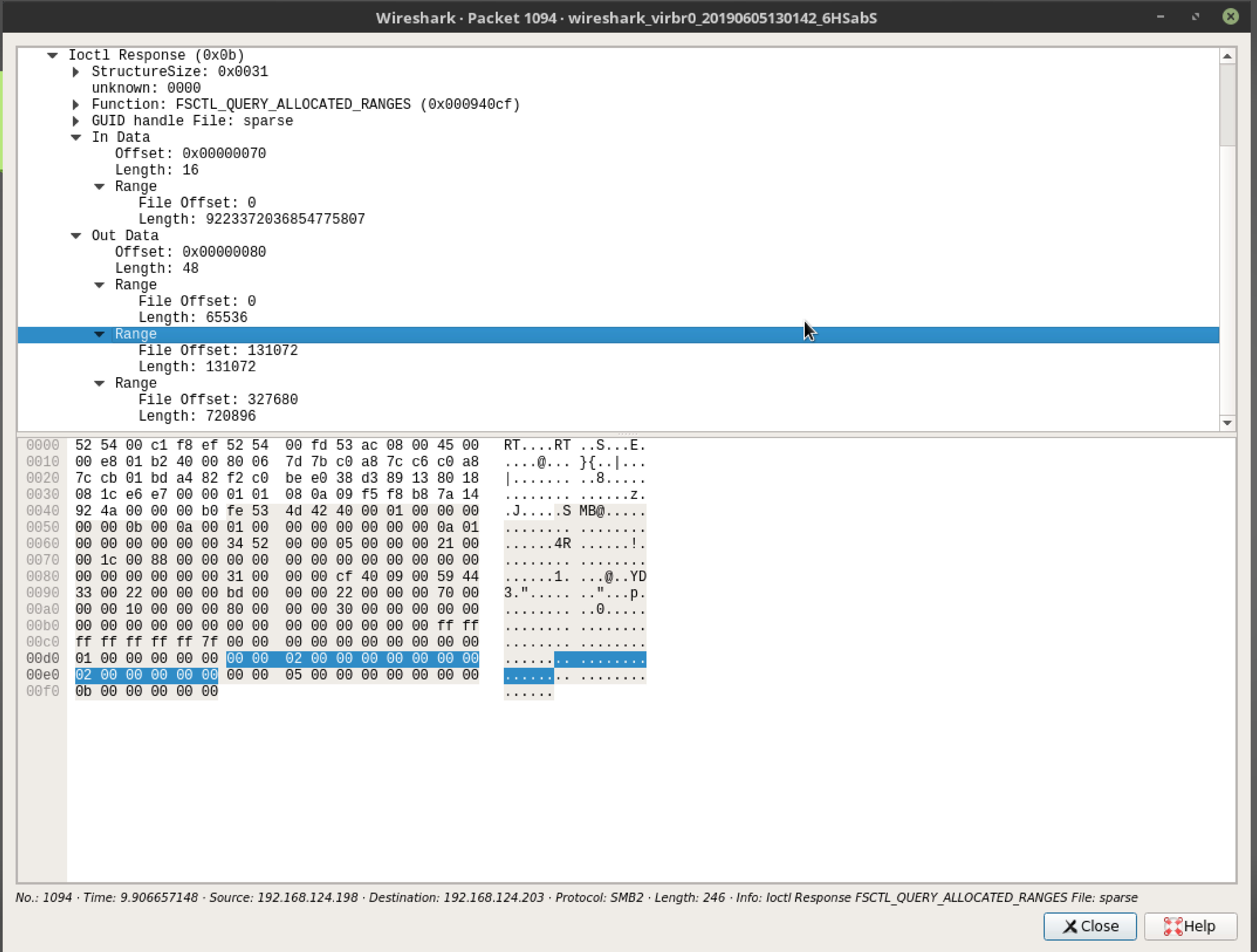
Compounding helps a lot – thanks Ronnie!

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

Sparse File Support (and other network fs can’t do this)!



```
[sahlberg@rawhide-2 cifs]$ #Create a sparse file
[sahlberg@rawhide-2 cifs]$ sudo ./sparse-file.py /mnt/sparse
Blocksize is 16384.
Changing this to 64k as that is real block size on windows16 Needs fixing.
0...65536
131072...262144
327680...1048576
[sahlberg@rawhide-2 cifs]$ #Check the FIEMAP
[sahlberg@rawhide-2 cifs]$ filefrag -v /mnt/sparse
Filesystem type is: fe534d42
File size of /mnt/sparse is 1048576 (64 blocks of 16384 bytes)
 ext:     logical_offset:        physical_offset: length:   expected: flags:
   0:        0..       3:          0..         3:      4:
   1:        8..      15:          8..        15:      8:
   2:       20..      63:         20..        63:     44:             last,eof
/mnt/sparse: 1 extent found
[sahlberg@rawhide-2 cifs]$ 
| [2 sahlberg@rawhide-2:/data/linux/fs/cifs] 0 sahlberg@rawhide-2:/  1 sahlberg@
```



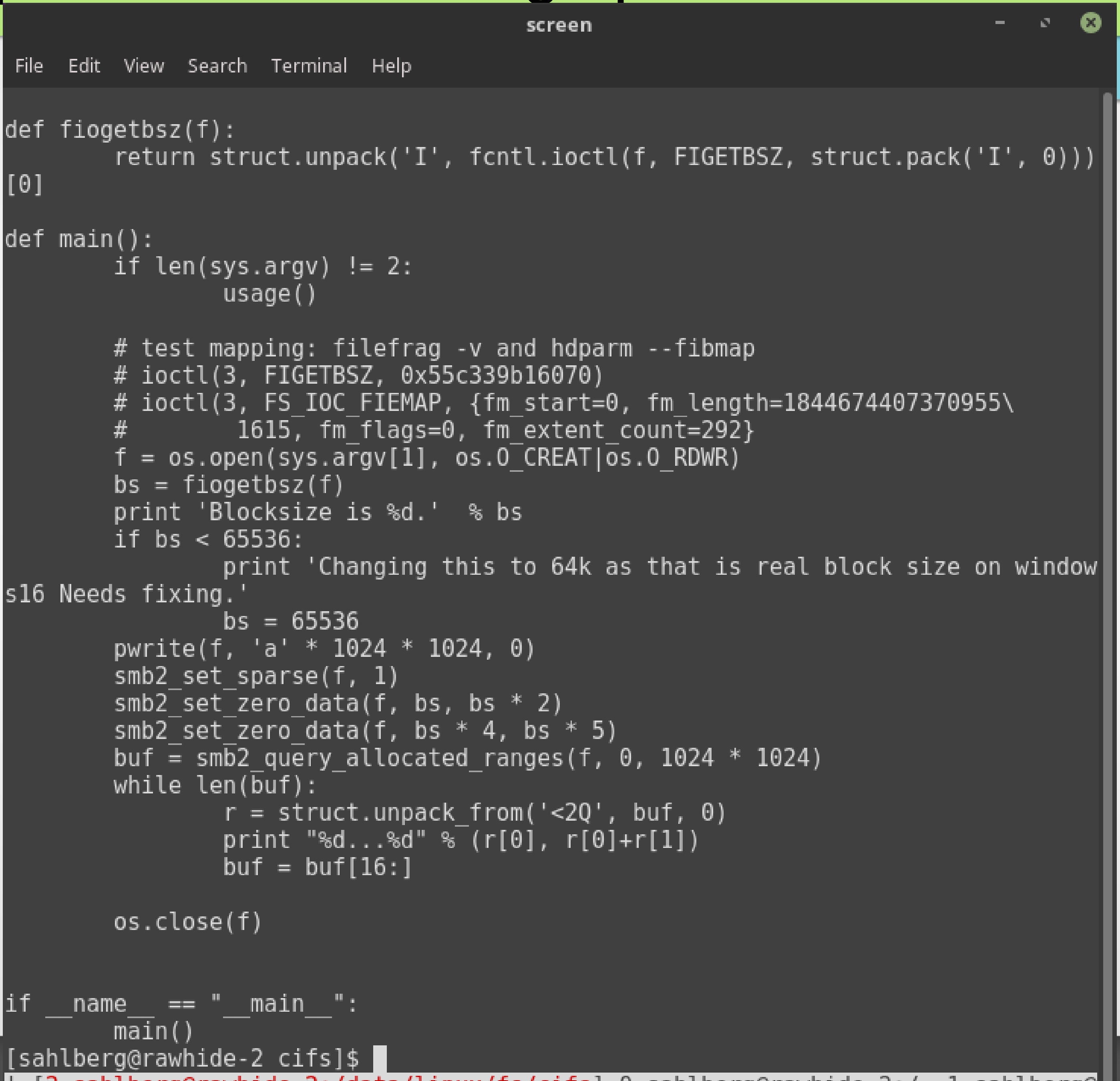
Wireshark · Packet 1094 · wireshark_virbr0_20190605130142_6HSabS

```
▼ Ioctl Response (0x0b)
  ▶ StructureSize: 0x0031
    unknown: 0000
  ▶ Function: FSCTL_QUERY_ALLOCATED_RANGES (0x000940cf)
  ▶ GUID handle File: sparse
  ▼ In Data
      Offset: 0x00000070
      Length: 16
    ▼ Range
        File Offset: 0
        Length: 9223372036854775807
  ▼ Out Data
      Offset: 0x00000080
      Length: 48
    ▼ Range
        File Offset: 0
        Length: 65536
    ▼ Range
        File Offset: 131072
        Length: 131072
    ▼ Range
        File Offset: 327680
        Length: 720896
```

```
0000  52 54 00 c1 f8 ef 52 54  00 fd 53 ac 08 00 45 00   RT....RT ..S...E.
0010  00 e8 01 b2 40 00 80 06  7d 7b c0 a8 7c c6 c0 a8   ....@... }{..|...
0020  7c cb 01 bd a4 82 f2 c0  be e0 38 d3 89 13 80 18   |....... ..8.....
0030  08 1c e6 e7 00 00 01 01  08 0a 09 f5 f8 b8 7a 14   ........ ......z.
0040  92 4a 00 00 00 b0 fe 53  4d 42 40 00 01 00 00 00   .J.....S MB@.....
0050  00 00 0b 00 0a 00 01 00  00 00 00 00 00 00 0a 01   ........ ........
0060  00 00 00 00 00 00 34 52  00 00 05 00 00 00 21 00   ......4R ......!.
0070  00 1c 00 88 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0080  00 00 00 00 00 00 31 00  00 00 cf 40 09 00 59 44   ......1. ...@..YD
0090  33 00 22 00 00 00 bd 00  00 00 22 00 00 00 70 00   3."..... .."...p.
00a0  00 00 10 00 00 00 80 00  00 00 30 00 00 00 00 00   ........ ..0.....
00b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 ff ff   ........ ........
00c0  ff ff ff ff ff 7f 00 00  00 00 00 00 00 00 00 00   ........ ........
00d0  01 00 00 00 00 00 00 00  02 00 00 00 00 00 00 00   ........ ........
00e0  02 00 00 00 00 00 00 00  05 00 00 00 00 00 00 00   ........ ........
00f0  0b 00 00 00 00 00                                  ......
```

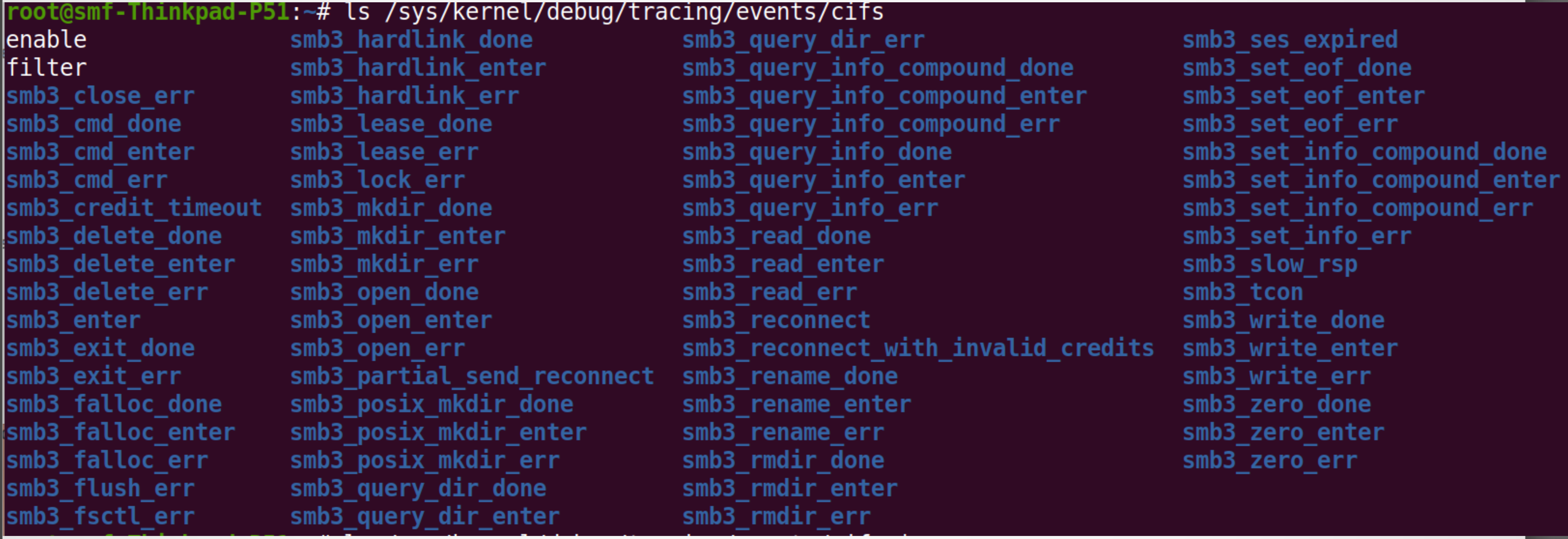
No.: 1094 · Time: 9.906657148 · Source: 192.168.124.198 · Destination: 192.168.124.203 · Protocol: SMB2 · Length: 246 · Info: Ioctl Response FSCTL_QUERY_ALLOCATED_RANGES File: sparse

Example test code using sparse files:



```
def fiogetbsz(f):
        return struct.unpack('I', fcntl.ioctl(f, FIGETBSZ, struct.pack('I', 0)))
[0]

def main():
        if len(sys.argv) != 2:
                usage()

        # test mapping: filefrag -v and hdparm --fibmap
        # ioctl(3, FIGETBSZ, 0x55c339b16070)
        # ioctl(3, FS_IOC_FIEMAP, {fm_start=0, fm_length=1844674407370955\
        #        1615, fm_flags=0, fm_extent_count=292}
        f = os.open(sys.argv[1], os.O_CREAT|os.O_RDWR)
        bs = fiogetbsz(f)
        print 'Blocksize is %d.'  % bs
        if bs < 65536:
                print 'Changing this to 64k as that is real block size on window
s16 Needs fixing.'
                bs = 65536
        pwrite(f, 'a' * 1024 * 1024, 0)
        smb2_set_sparse(f, 1)
        smb2_set_zero_data(f, bs, bs * 2)
        smb2_set_zero_data(f, bs * 4, bs * 5)
        buf = smb2_query_allocated_ranges(f, 0, 1024 * 1024)
        while len(buf):
                r = struct.unpack_from('<2Q', buf, 0)
                print "%d...%d" % (r[0], r[0]+r[1])
                buf = buf[16:]

        os.close(f)


if __name__ == "__main__":
        main()
[sahlberg@rawhide-2 cifs]$
```

Now 77 smb3 dynamic tracepoints (a year ago was 20 ...)!



```
root@smf-Thinkpad-P51:~# ls /sys/kernel/debug/tracing/events/cifs
enable               smb3_hardlink_done           smb3_query_dir_err                   smb3_ses_expired
filter               smb3_hardlink_enter          smb3_query_info_compound_done        smb3_set_eof_done
smb3_close_err       smb3_hardlink_err            smb3_query_info_compound_enter       smb3_set_eof_enter
smb3_cmd_done        smb3_lease_done              smb3_query_info_compound_err         smb3_set_eof_err
smb3_cmd_enter       smb3_lease_err               smb3_query_info_done                 smb3_set_info_compound_done
smb3_cmd_err         smb3_lock_err                smb3_query_info_enter                smb3_set_info_compound_enter
smb3_credit_timeout  smb3_mkdir_done              smb3_query_info_err                  smb3_set_info_compound_err
smb3_delete_done     smb3_mkdir_enter             smb3_read_done                       smb3_set_info_err
smb3_delete_enter    smb3_mkdir_err               smb3_read_enter                      smb3_slow_rsp
smb3_delete_err      smb3_open_done               smb3_read_err                        smb3_tcon
smb3_enter           smb3_open_enter              smb3_reconnect                       smb3_write_done
smb3_exit_done       smb3_open_err                smb3_reconnect_with_invalid_credits  smb3_write_enter
smb3_exit_err        smb3_partial_send_reconnect  smb3_rename_done                     smb3_write_err
smb3_falloc_done     smb3_posix_mkdir_done        smb3_rename_enter                    smb3_zero_done
smb3_falloc_enter    smb3_posix_mkdir_enter       smb3_rename_err                      smb3_zero_enter
smb3_falloc_err      smb3_posix_mkdir_err         smb3_rmdir_done                      smb3_zero_err
smb3_flush_err       smb3_query_dir_done          smb3_rmdir_enter
smb3_fsctl_err       smb3_query_dir_enter         smb3_rmdir_err
```

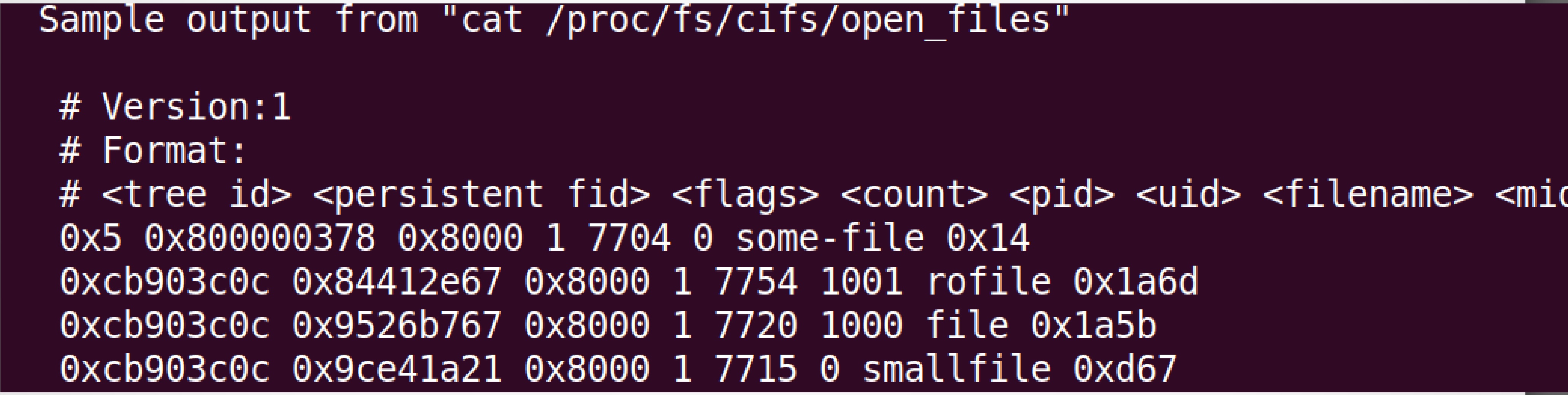
GCM Fast

- Can more than double write perf! 80% for read
- In 5.3 kernel

NetName context added!

- In 5.3 kernel

Can now view detailed info on open files (not just “lsof” output)

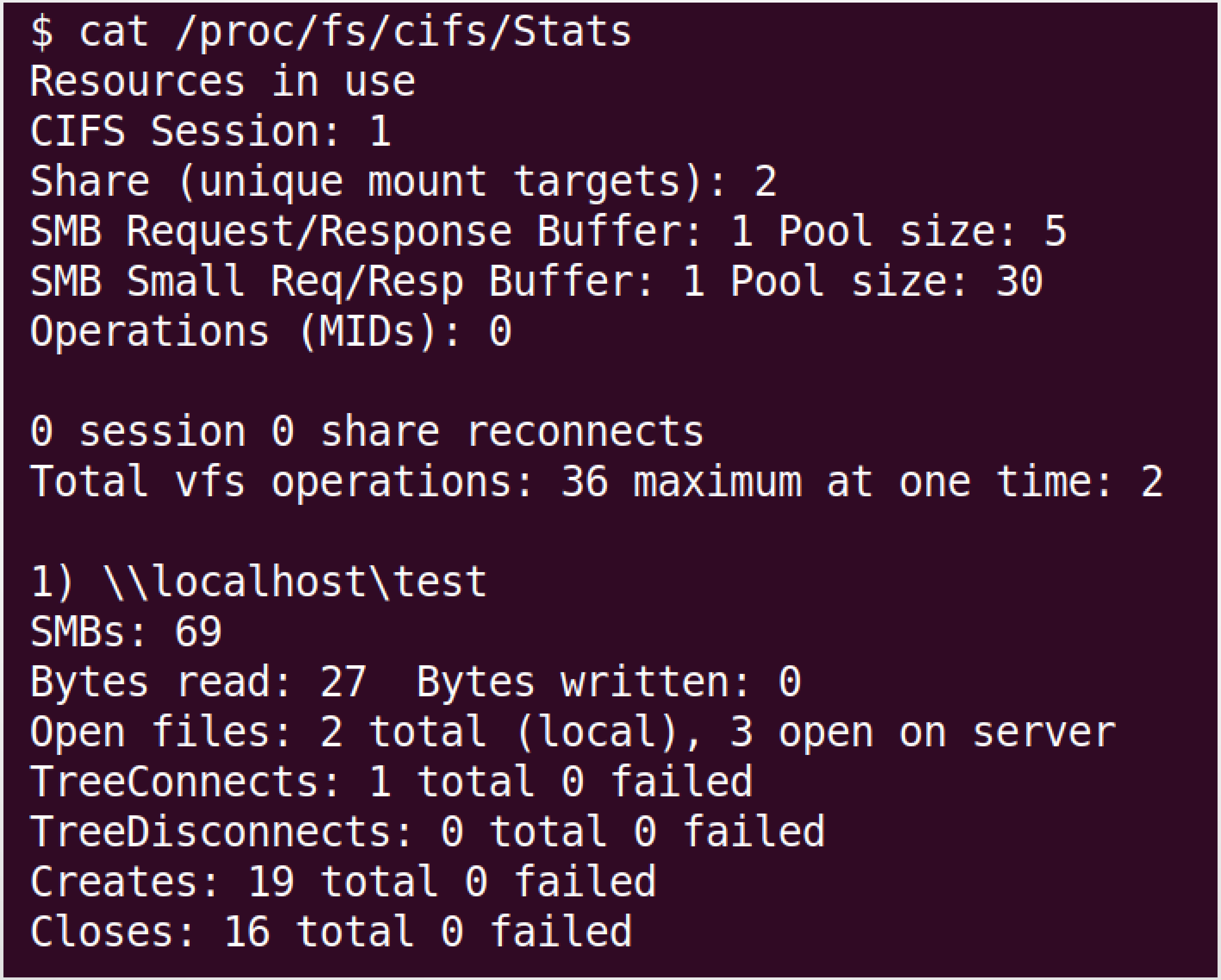


```
Sample output from "cat /proc/fs/cifs/open_files"

 # Version:1
 # Format:
 # <tree id> <persistent fid> <flags> <count> <pid> <uid> <filename> <mic
 0x5 0x800000378 0x8000 1 7704 0 some-file 0x14
 0xcb903c0c 0x84412e67 0x8000 1 7754 1001 rofile 0x1a6d
 0xcb903c0c 0x9526b767 0x8000 1 7720 1000 file 0x1a5b
 0xcb903c0c 0x9ce41a21 0x8000 1 7715 0 smallfile 0xd67
```

RDMA – Performance Improved

- Thank you Long Li! Many fixes/improvements
- No longer CONFIG_EXPERIMENTAL for smbdirect (RDMA) on Linux kernel client

And can view (on the client) the number of open files per share



```
$ 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
Operations (MIDs): 0

0 session 0 share reconnects
Total vfs operations: 36 maximum at one time: 2

1) \\localhost\test
SMBs: 69
Bytes read: 27  Bytes written: 0
Open files: 2 total (local), 3 open on server
TreeConnects: 1 total 0 failed
TreeDisconnects: 0 total 0 failed
Creates: 19 total 0 failed
Closes: 16 total 0 failed
```

Slow Response Threshold now configurable

/sys/module/cifs/parameters/slow_rsp_threshold

or via modprobe:

slow_rsp_threshold:Amount of time (in seconds) to wait before logging that a response is delayed.

Default: 1 (if set to 0 disables msg). (uint)

Recommended values are 0 (disabled) to 32767 (9 hours) with the default remaining as 1 second

SMB3/CIFS Features by release

- 4.17 (56 changesets) - June 3
  - Add signing support for smbdirect
  - Add support for SMB3.11 encryption, and preauth integrity (Thanks Aurelien!)
  - SMB3.11 dialect improvements (reconnect and encryption) (and no longer marked experimental)
- 4.18 (89 changesets) – August 12
  - RDMA and Direct I/O improvements (thank you Long Li!)
  - SMB3 POSIX extensions (initial minimal set, open and negotiate context only. use ‘posix’ mnt parm)
  - Add “smb3” alias to cifs.ko (“insmod smb3”)
  - Allow disabling less secure dialects through new module install parm (disable_legacy_dialects)
  - Add support for improved tracing (ftrace, trace-cmd)
  - Cache root file handle, reducing redundant opens, improving perf (Thanks Ronnie)

4.19 (69 cifs/smb3 changesets, Oct 22) cifs.ko internal module version 2.12

- Snapshot (previous version support)
- SMB3.1.1 ACL support
- Remove SMB3.1.1 config option (so support for newest dialect, SMB3.1.1, always built)
- Compounding for statfs (perf improvement)
- smb3 stats and tracepoints much improved (e.g. slow command trancepoint, counter)
- 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, lease break, security)

4.20 (70 Changesets, December 23rd) cifs.ko internal module version 2.14

- RDMA and direct i/o performance improvements (add direct i/o to smb3 file ops)
- Much better compounding (create/delete/set/unlink/mkdir/rmdir etc.), huge perf improvements for metadata access
- Additional dynamic (ftrace) tracepoints
- Add */proc/fs/cifs/open_files* to allow easier debugging
- Slow response threshold is now configurable
- 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.), and allow ioctl on directories
- Many Bug Fixes (including for krb5 mounts to Azure, and fix for OFD locks, backup intent mounts)

5.0 (82 changesets) March 3rd, 2019 (cifs.ko internal module version 2.17)

- SMB3.1.1 requested by default (ie is now in default dialects list)
- DFS failover support added (can reconnect to alternate DFS target) for higher availability and

  DFS referral caching now possible, cache updated regularly (Thank you Paulo)
- Support for reconnect if server IP address changes (coreq change in user space implemented in latest version of cifs-utils) (Thank you Paulo!)
- Performance improvement for get/set xattr (compounding support extended)
- Many Bug Fixes (24 important enough for stable) including for large file copy in cases where network connection is slow or interrupted, reconnect fixes, and fix for OFD lock support. The buildbot is really helping improve cifs.ko code quality!

5.1 (86 changesets) May 5th 2019 (cifs.ko internal module version 2.19)

- “fsctl passthrough” support improved: allows tools like cifs-utils to easily query any info available over SMB3 fsctl or query_info
- New mount parm “handletimeout” to allow persistent/resilient handle behavior to be configurable
- Allow fallocate zero range to expand a file
- Improve perf: cache FILE_ALL_INFO for the shared root handle
- Improve perf: default inode block size reported as 1MB (NB: 4MB rsize/wsize)
- Cleanup mknod and special file handling (thank you Aurelien)
- Support guest mounts over smb3.1.1
- Add many dynamic trace points to ease debugging and perf analysis
- Bug fixes (23 important enough for stable). Adding even more tests to the buildbot really helped. Multiple fixes for ‘crediting’ (SMB3 flow control) – thank you Pavel!

5.2 (64 changesets, so far) July 7 2019. cifs.ko version 2.20

- Bug fixes (11 important enough for stable)
- Improved perf: sparse file support now allows fiemap, SEEK_HOLE and SEEK_DATA (helps cp to Samba e.g.)
- Add support for fallocate ZERO_RANGE
- Support “fsctl passthrough” for cases where send (write) data in SMB3 fsctl – allows user space tools to do more!
- RDMA support (smbdirect) improved (thanks Long Li) should no longer be considered ‘experimental’

5.3 (55 changesets) Sept 15th, 2019 (cifs internal module number 2.22)

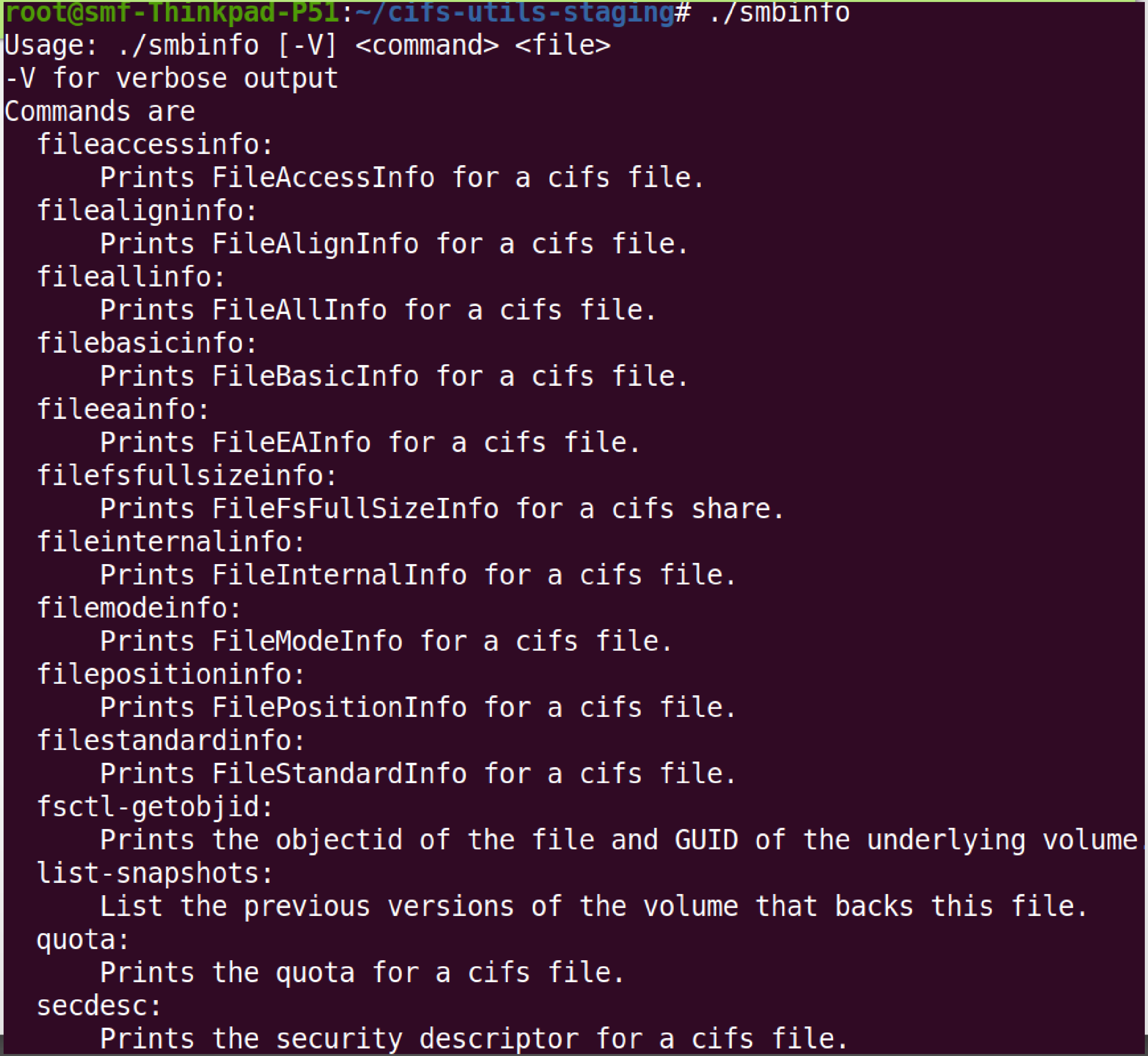
5.4 (38 changesets so far). Cifs version 2.23

Bugzilla

- Bugzilla.samba.org: 59 bugs open against cifsVFS component
- Bugzilla.kernel.org: 37 bugs open against CIFS component (down significantly)
- Help would be appreciated cleaning up some of these (some are already fixed but not backported by the distros they are using)

Cifs-utils enters a new era!

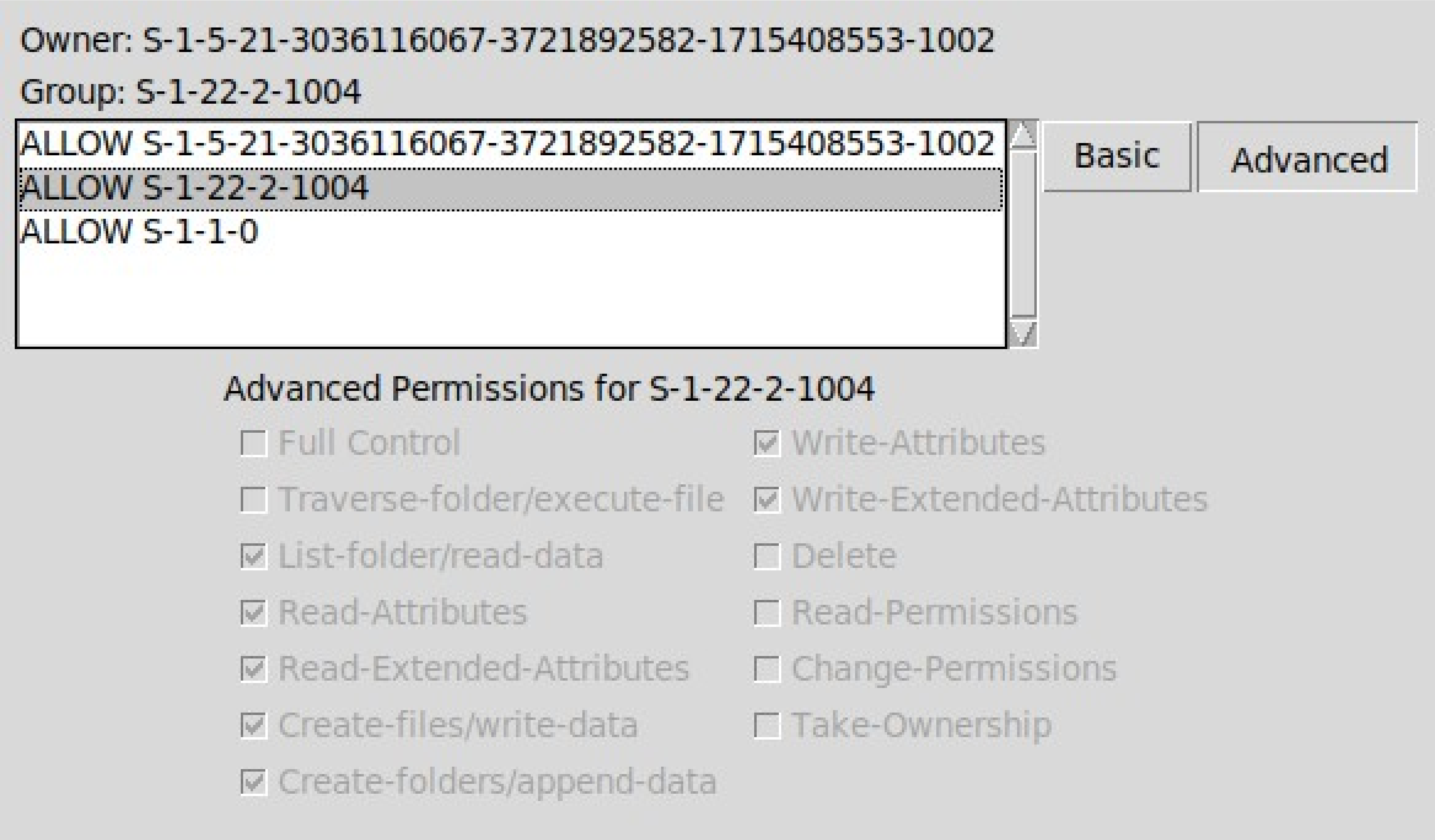
- Smbinfo!



```
root@smf-Thinkpad-P51:~/cifs-utils-staging# ./smbinfo
Usage: ./smbinfo [-V] <command> <file>
-V for verbose output
Commands are
  fileaccessinfo:
      Prints FileAccessInfo for a cifs file.
  filealigninfo:
      Prints FileAlignInfo for a cifs file.
  fileallinfo:
      Prints FileAllInfo for a cifs file.
  filebasicinfo:
      Prints FileBasicInfo for a cifs file.
  fileeainfo:
      Prints FileEAInfo for a cifs file.
  filefsfullsizeinfo:
      Prints FileFsFullSizeInfo for a cifs share.
  fileinternalinfo:
      Prints FileInternalInfo for a cifs file.
  filemodeinfo:
      Prints FileModeInfo for a cifs file.
  filepositioninfo:
      Prints FilePositionInfo for a cifs file.
  filestandardinfo:
      Prints FileStandardInfo for a cifs file.
  fsctl-getobjid:
      Prints the objectid of the file and GUID of the underlying volume.
  list-snapshots:
      List the previous versions of the volume that backs this file.
  quota:
      Prints the quota for a cifs file.
  secdesc:
      Prints the security descriptor for a cifs file.
```

Cifs-utils now even has a GUI!

- secddesc-ui.py



Owner: S-1-5-21-3036116067-3721892582-1715408553-1002

Group: S-1-22-2-1004

ALLOW S-1-5-21-3036116067-3721892582-1715408553-1002
ALLOW S-1-22-2-1004
ALLOW S-1-1-0

Advanced Permissions for S-1-22-2-1004

☐ Full Control
☐ Traverse-folder/execute-file
☑ List-folder/read-data
☑ Read-Attributes
☑ Read-Extended-Attributes
☑ Create-files/write-data
☑ Create-folders/append-data

☑ Write-Attributes
☑ Write-Extended-Attributes
☐ Delete
☐ Read-Permissions
☐ Change-Permissions
☐ Take-Ownership

Another example:

```
root@smf-Thinkpad-P51:~/cifs-utils-staging# ./smbinfo fileallinfo /mnt2/400M
Creation Time Fri May 24 15:06:42 2019
Last Access Time Fri May 24 15:06:42 2019
Last Write Time Fri May 24 15:06:43 2019
Last Change Time Fri May 24 15:06:43 2019
File Attributes 0x00000020:
Allocation Size 409993216
End Of File 409600000
Number Of Links 1
Delete Pending 0
Delete Directory 0
Index Number 10354692
Ea Size 0
File/Printer access flags 0x00020080:
Current Byte Offset 0
Mode 0x00000000:
File alignment: BYTE_ALIGNMENT
```

cifs-utils

- With pass-through SMB3 fsctl and query-info (and set-info) now possible it is easy to write user space tools to get any interesting info from the server
- Would love more contributions!
- Look at smbinfo from cifs-utils for examples

Features expected soon

- Security
  - Make the two common security use cases easy!
  - Improved GCM support (best SMB3.1.1 crypto, faster)
- Performance
- New function
  - Two new negotiate contexts:
    - Compression (partially implemented)
    - NETNAME_NEGOTIATE_CONTEXT
- More improvements to the SMB3.1.1 POSIX Extensions (see slides from yesterday)

Exciting year!!

- Faster performance
- Much better tools
- SMB3.1.1 default, improved security
- SMB3.1.1 POSIX Extensions improved!
- LOTS of new features
- And most importantly MUCH better tested and stable!

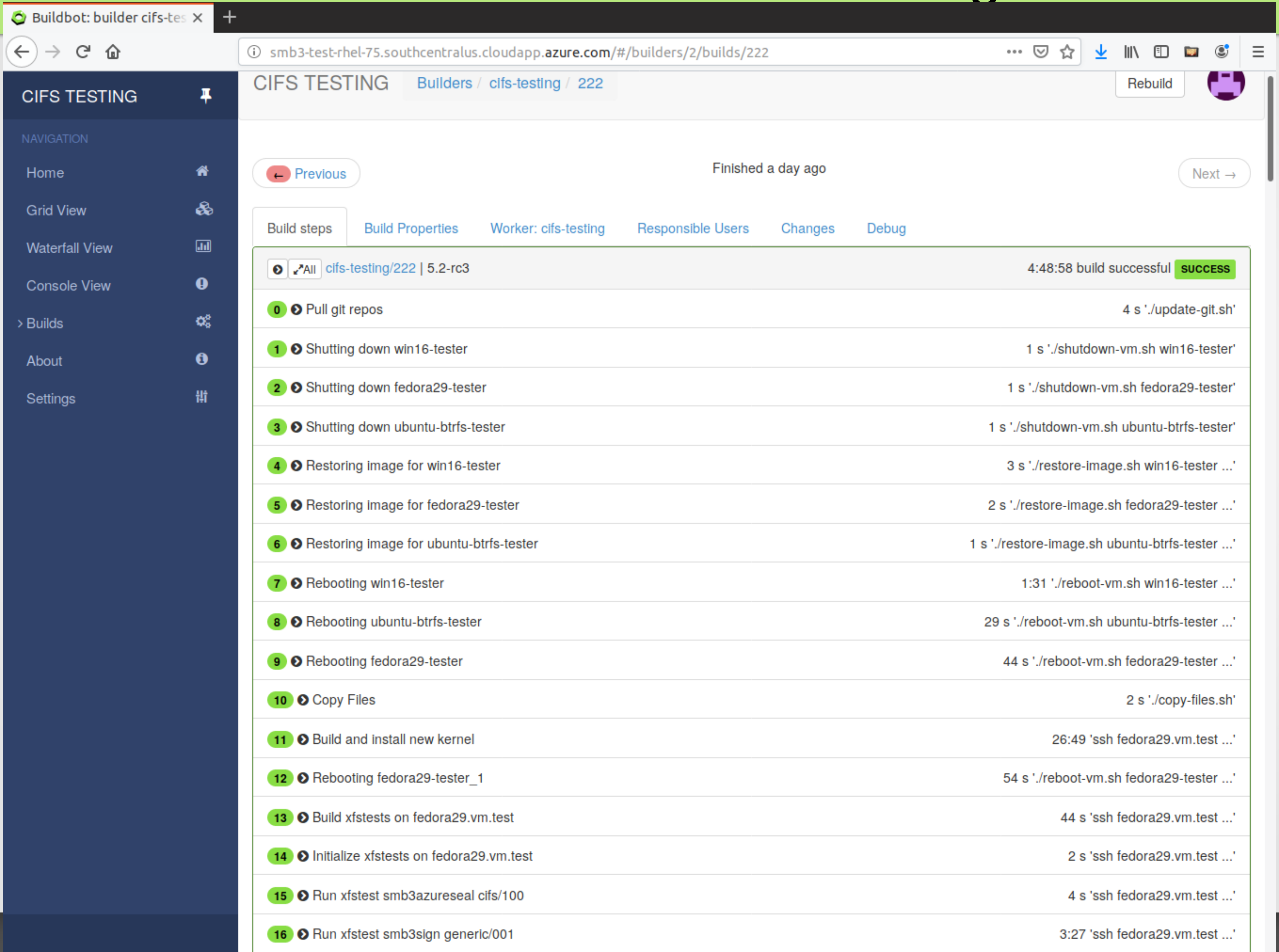
...

Testing … testing … testing

- The “buildbot” - automated regression testing! Thank you Paulo, Ronnie and Aurelien. See:

  http://smb3-test-rhel-75.southcentralus.cloudapp.azure.com
- 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
- Huge improvement in XFSTEST – up to 127 groups of tests run over SMB3 (more than run over NFS)! And more being added every release

The “buildbot” and automated testing



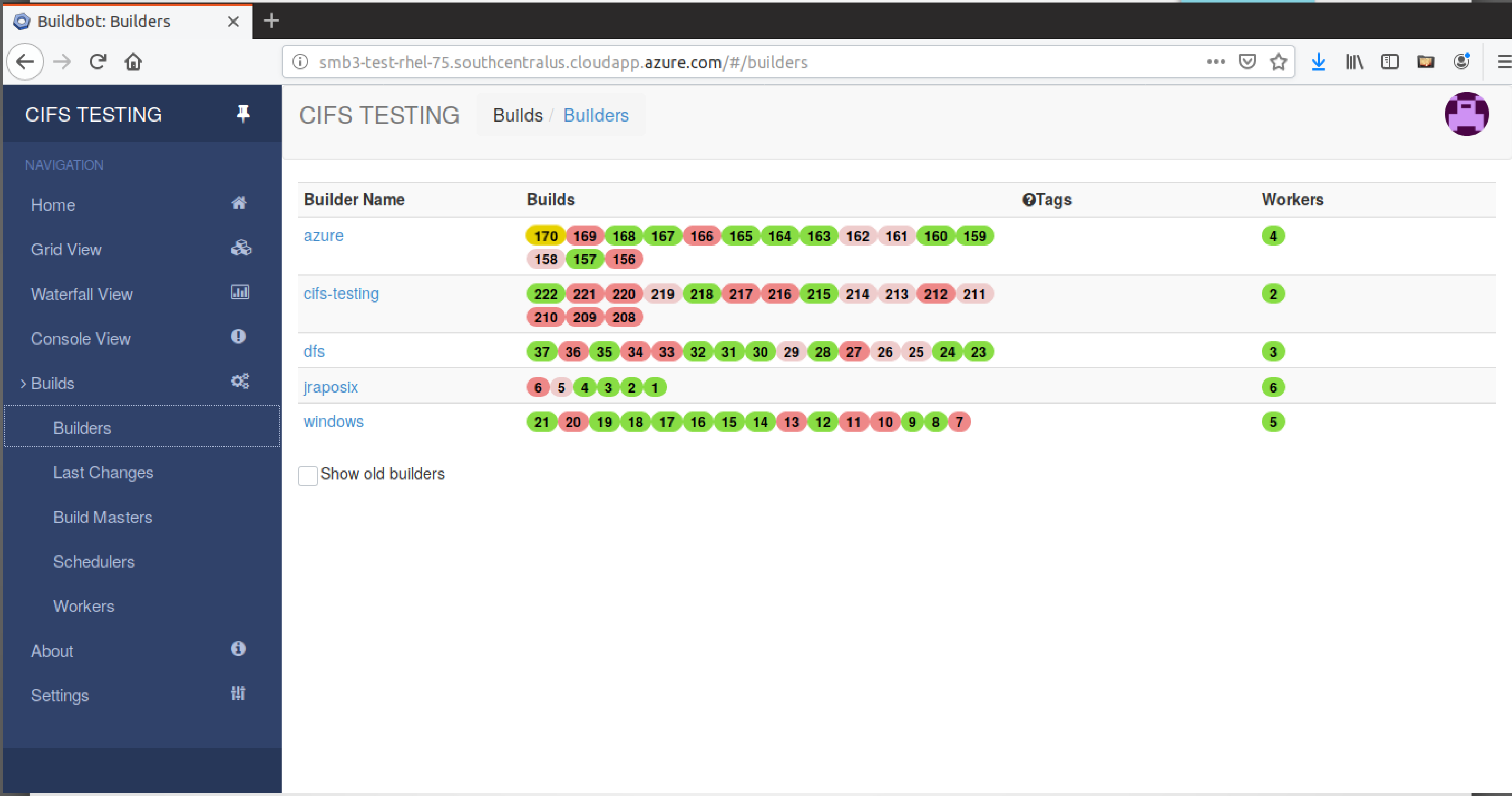
CIFS TESTING

Finished a day ago

cifs-testing/222 | 5.2-rc3 — 4:48:58 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	Shutting down fedora29-tester	1 s './shutdown-vm.sh fedora29-tester'
3	Shutting down ubuntu-btrfs-tester	1 s './shutdown-vm.sh ubuntu-btrfs-tester'
4	Restoring image for win16-tester	3 s './restore-image.sh win16-tester ...'
5	Restoring image for fedora29-tester	2 s './restore-image.sh fedora29-tester ...'
6	Restoring image for ubuntu-btrfs-tester	1 s './restore-image.sh ubuntu-btrfs-tester ...'
7	Rebooting win16-tester	1:31 './reboot-vm.sh win16-tester ...'
8	Rebooting ubuntu-btrfs-tester	29 s './reboot-vm.sh ubuntu-btrfs-tester ...'
9	Rebooting fedora29-tester	44 s './reboot-vm.sh fedora29-tester ...'
10	Copy Files	2 s './copy-files.sh'
11	Build and install new kernel	26:49 'ssh fedora29.vm.test ...'
12	Rebooting fedora29-tester_1	54 s './reboot-vm.sh fedora29-tester ...'
13	Build xfstests on fedora29.vm.test	44 s 'ssh fedora29.vm.test ...'
14	Initialize xfstests on fedora29.vm.test	2 s 'ssh fedora29.vm.test ...'
15	Run xfstest smb3azureseal cifs/100	4 s 'ssh fedora29.vm.test ...'
16	Run xfstest smb3sign generic/001	3:27 'ssh fedora29.vm.test ...'

Five test groups (should we add more?)



Builder Name	Builds	Tags	Workers
azure	170 169 168 167 166 165 164 163 162 161 160 159 158 157 156		4
cifs-testing	222 221 220 219 218 217 216 215 214 213 212 211 210 209 208		2
dfs	37 36 35 34 33 32 31 30 29 28 27 26 25 24 23		3
jraposix	6 5 4 3 2 1		6
windows	21 20 19 18 17 16 15 14 13 12 11 10 9 8 7		5

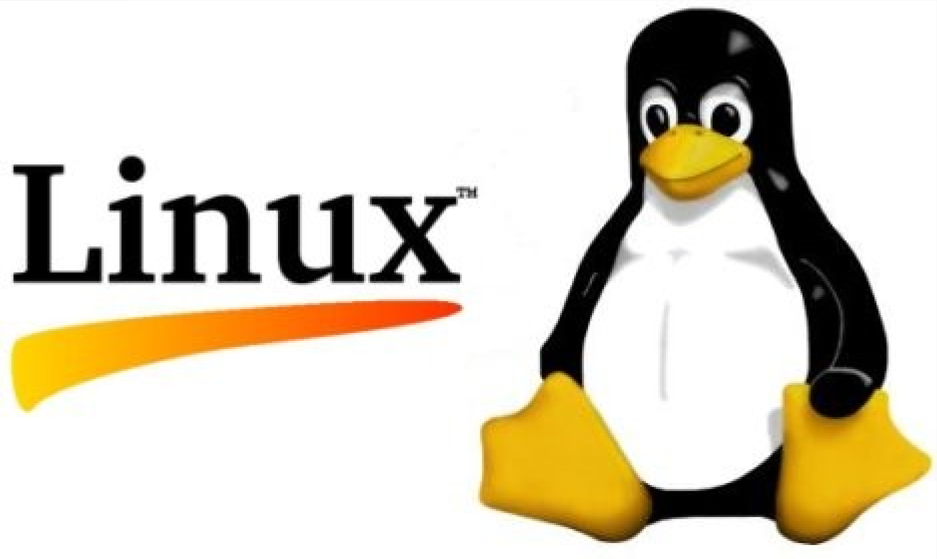
☐ Show old builders

Thanks to the buildbot – 5.1 Best Release Ever for SMB3!

- Prevents regressions
- Continues to improve quality

Thank you for your time

- Future is very bright!



+ S
M
B
3

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
  - https://wiki.samba.org/index.php/Xfstesting-cifs
  - 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