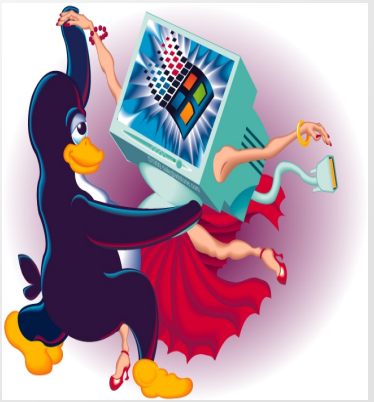
SMB3: Bringing High Performance File Access to Linux: A Status Update

How do you use it? What works?
What is coming soon?

Steve French
Principal Systems Engineer – Primary Data



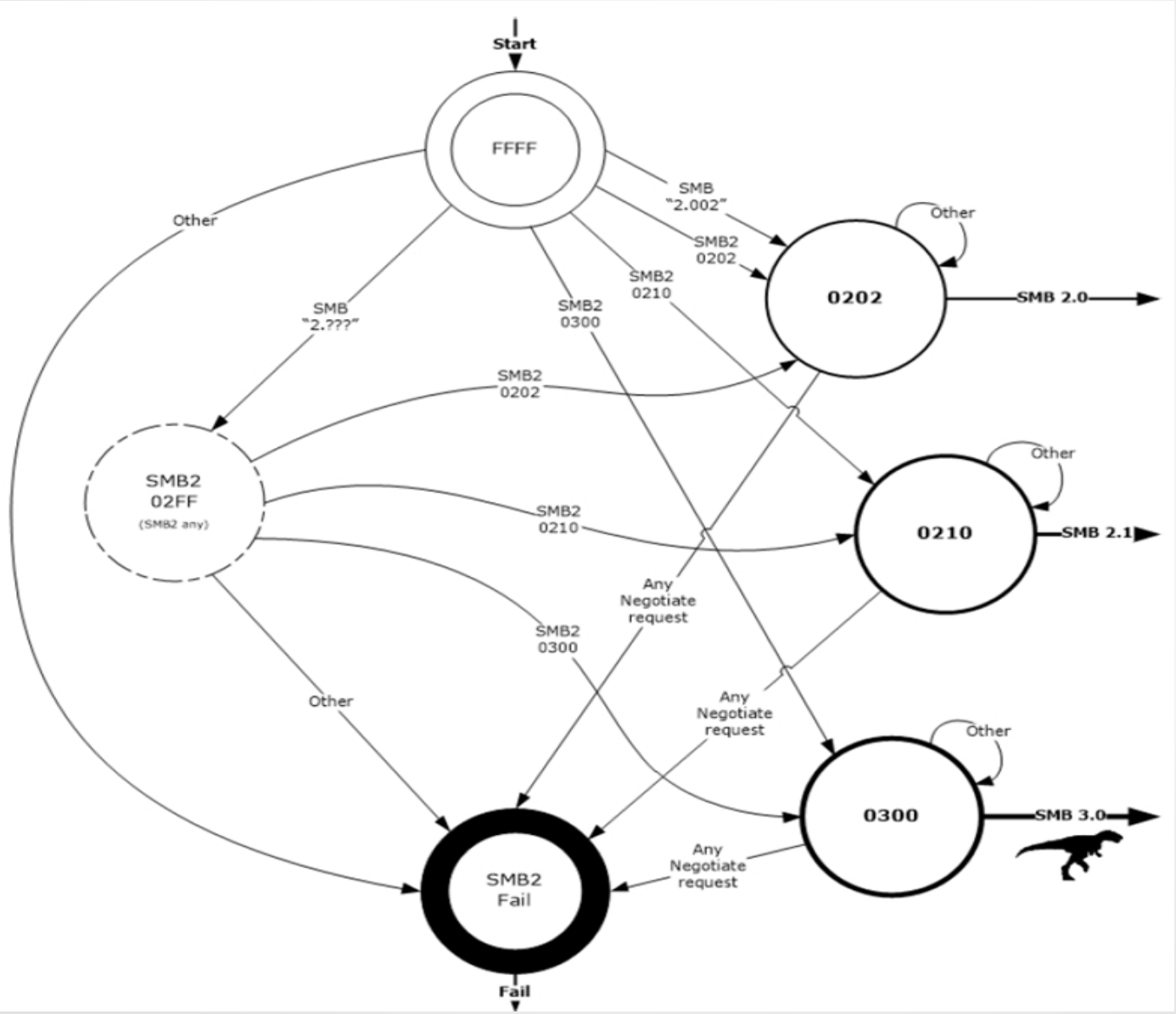
Legal Statement

- 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 SMB/CIFS based NAS appliances)
- Wrote initial SMB2 kernel client prototype
- Member of the Samba team, coauthor of SNIA CIFS Technical Reference and former SNIA CIFS Working Group chair
- Work for Primary Data

SMB3 Rocks


Start
FFFF
Other
SMB
"2.002"
SMB2
0202
SMB2
0210
SMB2
0300
SMB
"2.???"
0202
Other
SMB 2.0
SMB2
0202
SMB2
02FF
(SMB2 any)
0210
Other
SMB 2.1
SMB2
0210
Any
Negotiate
request
SMB2
0300
Other
Any
Negotiate
request
0300
Other
SMB 3.0
SMB2
Fail
Any
Negotiate
request
Fail

Development activity continues

- Kernel client (cifs.ko)
  - SMB2, 2.1 and 3.0 (and even minimal 3.02) support are in!
  - Current version is 2.03 and is visible via modinfo (and in /proc/fs/cifs/DebugData)
    - In one year we have gone from kernel 3.9 to 3.15-rc5
  - 210 kernel changesets for cifs, a typical year
  - More than 20 developers contributed
  - cifs continues to be one of the more active file systems in kernel
- Samba server also continues to improve its SMB2 and SMB3 support
  - And not just the server … Smbclient (user space ftp like tools) support SMB2

Kernel (including cifs client) improving

- A year ago we had 3.9 “Unicycling Gorilla”

- Now we have 3.15-rc5 “Shuffling Zombie Juror”

Features in process

- SMB3 Large i/o and multicredit perf improvements (Pavel)
- Auth cleanup, rewrite to improve gss auth support (Sachin)
- SMB3 ACL support
- Recovery of pending byte range locks after server failure (we already recover successful locks)
- Investigation into additional copy offload (server side copy) methods
- Full Linux xattr support
  - Empty xattr (name but no value)
  - Case sensitive xattr values
  - Security (SELinux) namespace (and others)
- SMB3 MF symlink support
- SMB3 Unix Extensions prototyping
- With Richard Sharpe's work on RDMA in the Samba server, is it time to push harder to do SMB3 RDMA on the kernel client?

Improvements by release

- 3.7 97 changes, cifs version 2.0
  - SMB2 added: **support for smb2.1 dialect added!**
  - remove support for deprecated "forcedirectio" and "strictcache" mount options
  - remove support for CIFS_IOC_CHECKUMOUNT ioctl
- 3.8 60 changes, cifs version 2.0
  - ntlmv2 auth becomes default auth (actually ntlmv2 encapsulated in NTLMSSP)
  - **smb2.02 dialect support added** and smb3 negotiation fixed
  - don't override the uid/gid in getattr when cifsacl is enabled
- 3.9 38 changes, cifs version 2.0
  - dfs security negotiation bug fixes (krb5 security). Rename fixes
- 3.10 18 changes, cifs version 2.01
  - cifs module size reduced
  - nosharesock mount option added
- 3.11 69 changes, cifs version 2.01
  - Various bug fixes: DFS, and workarounds for servers which provide bad nlink value
  - Security improvements (including SMB3 signing, but not SMB3 multiuser)
  - Auth and security settings config overhaul (thank you Jeff!)
  - SMB2 durable handle support (thank you Pavel!)
  - Minimal SMB3.02 dialect support

Improvements by release (continued)

- 3.12 40 changes, cifs version 2.02: **SMB3 support much improved**
  - SMB3 multiuser signing improvements, (thank you Shirish!) allows per-user signing keys on ses
  - SMB2/3 symlink support (can follow Windows symlinks)
  - Lease improvements (thank you Pavel!)
  - debugging improvements
- 3.13 34 changes
  - Add support for setting (and getting) per-file compression (e.g. “chattr +c /mnt/filename”)
  - Add SMB copy offload ioctl (CopyChunk) for very fast server side copy
  - Add secure negotiate support (protect SMB3 mounts against downgrade attacks)
  - Bugfixes (including for setfacl and reparse point/symlink fixes)
  - Allow for O_DIRECT opens on directio (cache=none) mounts. Helps apps that require directio such as newer specsfs benchmark and some databases
  - Server network adapter and disk/alignment/sector info now visible in /proc/fs/cifs/DebugData
- 3.14 27 changes
  - Security fix for make sure we don't send illegal length when passed invalid iovec or one with invalid lengths
  - Bug fixes (SMB3 large write and various stability fixes) and aio write and also fix DFS referrals when mounted with Unix extensions

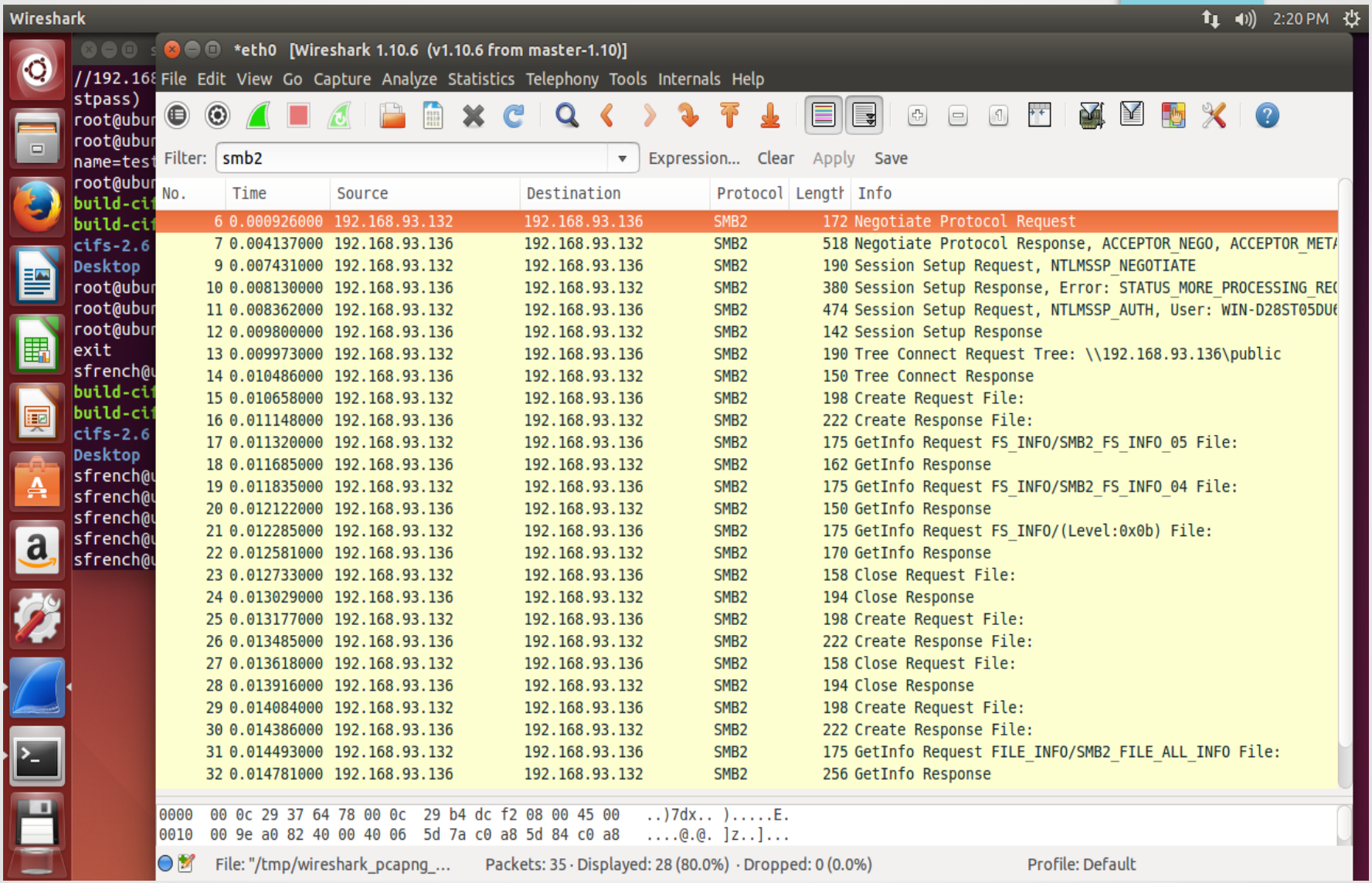
Improvements by release (continued)

- 3.15 17 changes
  - Various minor bug fixes (include aio/write, append, xattr, and also in metadata caching)
- Changes planned for 3.16 (or soon thereafter)
  - Allow multiple mounts to same server with different dialects
  - Authentication session establishment rewrite to improve gssapi support
- 3.17 plan to add higher performance large read/write, SMB2/SMB3 multicredit support

Cifs-utils

- The userspace utils: mount.cifs, cifs.upcall,set/getcifsacl,cifscreds, idmapwb,pam_cifscreds
  - thanks to Jeff Layton for maintaining cifs-utils
- 31 changesets over the past year
  - Current version is 6.3.1
  - Includes various bugfixes (especially in setcifsacl util)
  - Dedicated kerberos keytab (other than system default) can be specified.
- Also of note: in 12/2012 Idmap plugin supportwas added (allows sssd, not just winbind, cached userid information to be used) in version 5.9 of cifs-utils

SMB3.02 Mount to Windows



SMB3 Kernel Client Status

- SMB3 support is solid, but lacks many optional features
- Can mount with SMB2.02, SMB2.1, SMB3, SMB3.02
  - Specify vers=2.0 or vers=2.1 or 3.0 or 3.02 on mount
  - Default is cifs but also mounting with vers=1.0 also forces using smb/cifs protocol
  - Default will change to SMB3 when Unix extensions available for SMB3, and performance and functional testing is as good or better

SMB3 Kernel Status continued

- In:
  - SMB2.1 Lease support (improved caching)
  - SMB2 durable handles (improved data integrity)
  - SMB3 signing (including for multiuser mounts)
    - Downgrade attack protection (secure negotiate)
  - Dynamic crediting (flow control)
  - Not SMB3 specific: Compressed files, copy offload
  - Windows 'NFS' symlinks (partial)

SMB3 Kernel Status continued

- TODO
  - ACLs for SMB2/SMB3
  - 3 types symlinks: Windows, Windows 'NFS' and 'MF"
  - POSIX/Unix extensions (see recent work by Volker)
  - Optional features:
    - Multichannel (started) and RDMA
    - Persistent handles
    - Witness protocol, improved cluster reconnection
    - Encrypted share support
    - ODX Copy Offload support (but can do CopyChunk)
    - Large Read/Write support (in progress) and compound ops

SMB POSIX Extensions

POSIX Behaviors requested on TreeConnect in current CIFS Unix Extensions

Those likely still required are in bold

#define CIFS_UNIX_FCNTL_CAP 0x00000001 /* support for fcntl locks */

#define CIFS_UNIX_POSIX_ACL_CAP 0x00000002 /* support getfacl/setfacl */

#define CIFS_UNIX_XATTR_CAP 0x00000004 /* support new namespace */

#define CIFS_UNIX_EXTATTR_CAP 0x00000008 /* support chattr/chflag */

#define CIFS_UNIX_POSIX_PATHNAMES_CAP 0x00000010 /* Allow POSIX path chars */

#define CIFS_UNIX_POSIX_PATH_OPS_CAP 0x00000020 /* POSIX path based calls including posix open posix unlink */

#define CIFS_UNIX_LARGE_READ_CAP 0x00000040 /* support reads >128K (up to 0xFFFF00 */

#define CIFS_UNIX_LARGE_WRITE_CAP 0x00000080

#define CIFS_UNIX_TRANSPORT_ENCRYPTION_CAP 0x00000100 /* can do SPNEGO crypt */

#define CIFS_UNIX_TRANSPORT_ENCRYPTION_MANDATORY_CAP 0x00000200 /* must do */

#define CIFS_UNIX_PROXY_CAP 0x00000400 /* Proxy cap: 0xACE ioctl and QFS PROXY call */

SMB3 POSIX Extensions

- Existing FILE_UNIX_INFO_BASIC (from cifs unix extensions, **bold** are those needed in SMB3)
  - __le64 EndOfFile;
  - __le64 NumOfBytes;
  - __le64 LastStatusChange; /*SNIA specs DCE time for the 3 time fields */
  - __le64 LastAccessTime;
  - __le64 LastModificationTime;
  - **__le64 Uid;**
  - **__le64 Gid;**
  - **__le32 Type;**
  - **__le64 DevMajor;**
  - **__le64 DevMinor;**
  - **__le64 UniqueId; /* Unique ID can be requested in SMB3 but not returned on Open? */**
  - **__le64 Permissions;**
  - **__le64 Nlinks;**

SMB3 POSIX Extensions

- Unix/POSIX Extensions in CIFS were done via QueryInfo and FindFirst and FSInfo extensions (new infolevel) and negotiation of capabilities at tree connect time
- Presence of SMB3 POSIX Extensions could be advertise via FS Info, but data returned via SMB3 Create Context
  - Requested behaviors (posix locks, unlink, case) requested in Create Context as well

- Send requested POSIX capabilities, return those granted
  - POSIX_FCNTL_LOCK_CAP
  - POSIX_UNLINK_CAP (POSIX file behaviors)
  - POSIX_INODE_CAP (POSIX inode info like UID, mode)
  - POSIX_PATHNAMES_CAP
  - CASE_SENSITIVE_CAP
  - And perhaps also
    - **POSIX_ACL_CAP**
    - **CHATTR/CHFLAGS CAP**
    - **Support for Linux xattr alt namespaces (e.g. Trusted & Security for SELinux)**

SMB3 POSIX Extensions

- Opening a file
  - Request
    - posix_request_flags (do we need to send mandatory vs. optional indication?)
  - Return
    - posix_granted_flags
      -
  - Response (if posix info requested)
    - **__le64 Uid;**
    - **__le64 Gid;**
    - **__le32 Type;**
    - **__le64 DevMajor;**
    - **__le64 DevMinor;**
    - **__le64 UniqueId; /* Unique ID can be requested in SMB3 but not returned on Open? */**
    - **__le64 Permissions;**
    - **__le64 Nlinks;**

Testing … testing … testing

- One of the goals for this summer is to improve automated testing of cifs.ko
- Functional tests:
  - Xfstest is the standard file system test bucket for Linux
    - Runs over nfs, I created patch to run over cifs/smb3
      - Found multiple bugs when ran this first
    - Challenge to figure out which tests *should* work (since some tests are skipped when run over nfs and cifs)
  - Other functional tests include cthon, dbench, fsx
- Performance/scalability testing
  - Specsfs works over cifs mounts (performance testing)
  - Big recent improvements in scalability of dbench (which can run over mounts)
  - Various other linux perf fs tests work over cifs (iozone etc.)
  - Need to figure out how to get synergy with iostats/nfsstats/nfsometer

Thank you for your time