Opening Windows to a Wider World

What's new in Samba - 2020

Jeremy Allison
Samba Team
jra@samba.org
jra@google.com

How is the fileserver structured internally ?

- Three conceptual layers.
  - SMB1/2/3 protocol parsing layer.
  - NTFS (Windows NT Filesystem) emulation (making Linux act like Windows)
  - VFS – access to local and remote filesystems.
- In practical terms the separation of the upper two layers isn't so clean in the codebase.
  - At least for the SMB1 code.
  - SMB2+ layers are better designed.

Fileserver modernization

Fileserver TODO list.

- Removal of SMB1.
- Modernization of the VFS.
- More asynchronous internals.
  - Threading under the covers.
- Performance improvements.
  - Clustering improvements.
- Service framework improvements.
- SMB over QUIC (speculative).

“Friends don’t let friends use SMB1” - Ned Pyle (Microsoft)

- SMB1 enabled by default was removed in the Samba 4.11 release.
  - The code is still there, can be turned back on for older systems.
- SMB1 code will be removed for Samba 5.0.
  - Whenever we remove the code, that’s Samba 5.0 :-).
  - SMB1 code makes internal modernization / maintenance costly.
  - Lots of pathname-based operations.
  - UNIX extensions for SMB2 pending on VFS rewrite.

Modernizing the VFS

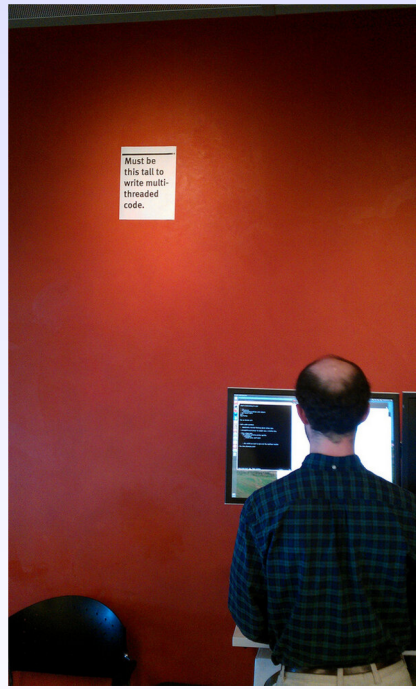
- Samba VFS was originally designed around POSIX in the 1990's.
  - open()/read()/write()/close() etc.
- Modern UNIX system calls are completely different.
  - openat(), fstatat(), rmdirat() etc.
  - Take relative file descriptor
  - Symlink-safe (if used correctly).
- Also the needs of MT-code (also credential impersonation) are not well served by the current VFS design.

Modernizing the VFS (continued)

- New XXXAT() VFS looks like:
  - SMB_VFS_MKDIRAT(handle, dirfsp, smb_fname, mode)
  - All names will be relative to dirfsp.
- Moving to be closer to NTFS (Windows) requirements.
  - Make easier for OEMs to plug in back-end cluster and advanced filesystems.
- Only partly done for 4.12 – should be completed by 4.13.
  - Some unavoidable churn for OEMs writing custom VFS modules.

Asynchronous Internals

- Mid-level (NTFS) code is still single threaded.
  - Move to async calls into the VFS to parallelize.
  - Multiple outstanding calls on the go at once.
  - Incoming / outgoing socket to client is still a synchronization point.
- New async VFS calls look like:
  - SMB_VFS_PREAD_SEND() / SMB_VFS_PREAD_RECV().
  - Allow Samba pthreadpool code to be used under the covers.
- New impersonation infrastructure in progress.



Performance improvements

- Moving to GnuTLS encryption code gives 3x speedup on encrypted connections (AES-CCM → AES-GCM).
- Lots of work done on internal databases.
  - XXX improvement on common case of share mode entry at root of share being opened by all mounting clients (see Volker's talk).
  - Careful examination and separation out of data models needed for Windows cluster semantics.
  - Samba gencache - Caching performance improvements.
- Lots of small scalability fixes added.
- Linux io_uring VFS module added for 4.12.
  - ~ 20% improvements in read

Clustering improvements

- Clustered Windows semantics (persistent handles) will always be hard / slow.
  - Every open has to check share modes across the cluster.
  - Data caching helps here.
- Plan for implementing persistent handles in Samba 4.x (x > 12).
- Many improvements in Samba ctdb cluster manager.
  - Continuous integration / testing under development.
- Goal is to get to plugable clustering. Decouple from ctdb to allow third party cluster managers to replicate Samba

Service improvements

- Home-grown crypto removed. Standardize on GnuTLS.
  - Old code served us well and allowed us to quickly iterate, but no one should write their own crypto.
- Insane RPC framework duplication removed.
  - Two RPC server implementations.
  - Two RPC client implementations.
  - RPC server framework merge – code being worked on in gitlab, not yet in master.
- Full async RPC calls close to merge.
  - Needed for SMB witness service.
- Work ongoing to allow RPC services to be proxied to

SMB3 over QUIC

- Microsoft have experimental servers / clients running on Windows.
  - Microsoft is happy to open protocol and document changes needed.
- Samba implementation is awaiting stable QUIC library framework and service manager framework on Linux we can plug into.
  - How do we route QUIC connection requests from web server to smbd ?
  - Lots of interest, but no code yet.
- SMB3 over QUIC is the future of SMB over the Internet and into Cloud storage.

Samba Active Directory

- Great number of performance / scaling improvements.
  - 300K users now feasible.
  - Prefork model adopted for most AD-service components.
- Supports smart card authentication.
- JSON audit / security logging.
- Gaining use in Government installations.
  - Some missing features, mostly around Active Directory Web Services.
  - Samba doesn't want to be in the Web-server business.

Modernizing the project infrastructure

- Gitlab / Continuous integration – where the cool kids are !
  - Project workflow has mostly moved to gitlab.
  - Project master code still held on samba.org, but much easier for external contributors.
- Continuous integration tests now easy – on every push.
  - 2 Samba-Team member engineer review needed for any external contribution, so extra work put on existing engineers not drive-by coders.
- Fuzzing
  - Initial fuzzing with Codenomicon (proprietary tool).

General Free Software SMB updates

- New LGPLv2 SMB2- only library added to Samba project. libsmb2
  - Tiny footprint (140kb) user-space client library.
  - No external dependencies (other than kerberos libraries).
  - Zero-copy for reads/writes (except for encrypted connections).
- Linux kernel may be getting an experimental in-kernel SMB2+ server - ksmbd.
  - Limited functionality as yet, but a project to watch !

Questions and Comments ?

Email: jra@samba.org
Email: jra@google.com