

Improving the SAMBA VFS for Zero Samba Mods

Richard Sharpe

Panzura

Agenda

- What do I mean by zero Samba mods
- Why do we care?
- What do OEMs do today?
- How can we get to zero mods?
- The changes that would be useful
- Going further

What do I mean by zero Samba mods

- OEMs can ship a product that uses Samba
 - Without having to modify Samba
- They might have to write one or more VFS modules

Note: There is already Wiki page on improving the VFS.

Why do we care

- Reduce the complexity of
 - Using Samba in a product
 - Supporting Samba in a product
 - Merging their patches and our patches is hard work
 - Upgrading to new versions of Samba
 - Which of their patches still apply? Cleanly?
- Makes for happier OEMs
 - More predictable transitions to new versions

What OEMs do today

- OEMs I have worked with
 - Panasas
 - Data Domain
 - Drobo
 - Scale Computing
 - Dell/Ocarina
 - Tandberg
 - Panzura

What do OEMs Do Today

- Panzura
 - At least 54 mods, but likely more
 - At least 5 VFS modules
- QNAP
- Synology
- Others

Some mods

- Correct Windows Previous Versions Behavior
 - Numerous changes
 - Return RO Volume in `smbd_do_qfsinfo`
 - Required an FSP be passed in
 - Must allow Write-like opens
 - Even though ROFS
 - Fall back to `O_RDONLY` on `open(2)`
 - Return `NT_STATUS_MEDIA_WRITE_PROTECTED`
 - When incompatible operation
 - Touched a lot of places

Some mods, cont

- Sometimes Samba too noisy
 - Dumps core when it is of not much use
 - The level 0 log is enough
 - Prints out errors at wrong level
 - Relatively minor issue
- Sometimes Samba not noisy enough
 - Silently converts errors to ACCESS_DENIED
 - Should log at level 0 so we can see these cases

Some mods, cont

- Did not want to call `ntimes` after file closed
 - Samba does that to ensure that times set by `SetInfo` do not change on `close(2)` (we think)
 - ZFS does not change times on `close`
 - Calling `ntimes` after `close` caused races

Some mods, cont

- Hiding things from readdir
- Exposing things to readdir/Find First/Query
 - ~snapshot
 - ~policy
 - Etc

Some mods, cont

- QueryDeleteOnClose

Some mods, Synology

- A point in time
 - Based on Samba 3.6.9 when I checked
 - A 2.9MB patch file!
 - Has their change group membership when signal
 - Plus others

Some mods, Tandberg

- Tandberg wanted the Synology fix
 - Doing it for real took longer than anticipated
 - It is harder than anticipated
 - Required changes to:
 - messaging.idl, srv_srvsvc.c, proto.h, sharesec.c
 - service.c, server.c, process.c, auth_util.c, token_util.c
 - New file auth_reload.c
 - Will send changes out soon for people to look at

Some mods, Drobo

- Some integration with netatalk
- Some build fixes
- Pretty much all available in recent builds

Some mods, Data Domain

- Had a user-space file system accessed via NFS
- Needed support for readdirplus
- Moved to Likewise now
 - However, that took around five years

How can we get to zero mods?

- If you simply ship Samba as is
 - No special VFS modules, etc
 - It is easy
 - Download the RPMs from SerNet
- Otherwise, an impossible goal
 - At least have to add your VFS module(s) to source tree
 - Change/add wscript files
 - Build

What can we do?

- Carefully figure out what is needed
 - Provide correct interfaces
 - Current VFS accreted over time
- Develop a plan to implement changes
 - Some are relatively simple
 - Some are much harder

Changes that would be useful

- VFS Changes
- Samba Changes
- Build changes

VFS Changes

- Always return NTSTATUS from VFS functions
- Pass an FSP to more functions
- Provide for locking calls to the VFS
- Push CTDB calls through a VFS

Return NTSTATUS from VFS Functions

- VFS started out with UNIX-style error handling
 - I have added new errno values
 - ESHARINGVIOLATION, etc
- Added some functions with NTSTATUS returns
 - As needed
- Needs to be consistent
 - Some have suggested an upper VFS and a lower VFS.

Pass an FSP to more functions

- ntimes
- directory scanning functions
- Why?
 - Access state stashed in an fsp extension
- Strategy
 - If you are not passing an fsp, justify it
 - Some cannot take an fsp however
 - connect
 - disconnect

Provide for locking calls

- Some VFS modules need to take locks
- To avoid deadlock, provide locking hooks
 - Separate from Byte Range Locks
- Needs justification

Push CTDB calls through a VFS

- All the CTDB calls are made via TDB wrappers
- Should be in a/the VFS
 - Then other mechanisms can be supported

Samba changes

- Design a cleaner VFS interface
 - Perhaps an upper and lower VFS
- This is harder to do, but is needed
 - However, we have reasonable tests in Samba 4
- OEMs do not need to understand the Samba code
 - Just the VFS interface

Build Changes

- With Samba 3.5 and 3.6
 - In tree or
 - Out of tree
- This is not possible currently with Samba 4
- Need to think about providing an include files package
 - OEMs do not need to build SAMBA and keep the source around

Going further

- Move VFS Developer Info to the Wiki
 - [Partially] Done
 - Will have a version differences section
 - Allows for multiple contributors
- Would like more input
 - From OEMs
 - From other VFS module developers
 - Send it to me
- Do the simple things first

Links

- The Samba VFS
 - https://wiki.samba.org/index.php/The_new_version_of_Writing_a_Samba_VFS
- Developer WIKI on the VFS:
 - https://wiki.samba.org/index.php/Samba3/SMB2#VFS_layering:_introduce_a_NT-FSA-layer