Lessons learned from using Samba in IBM Spectrum Scale

Christof Schmitt
christof.schmitt@us.ibm.com
cs@samba.org

IBM / Samba Team

sambaXP 2020

Table of Contents

1. Integrating Samba in IBM Spectrum Scale product
2. Examples of issues faced
3. Lessons learned

Abstract

IBM Spectrum Scale is a software defined storage offering of a clustered file system bundled together with other services. Samba is included as part of the product to provide a clustered SMB file server and integration into Active Directory. This talk discusses from a development point of view the integration of Samba into a storage product and what the development team has learned over the years. Topics will include the requirement for automated testing on multiple levels and the collaboration with the upstream Samba project. Examples will be used to illustrate problems encountered over time and how they have been solved. Further topics will be challenges that have been solved and gaps that have been seen with the usage of Samba.

About me

- Senior Software Engineer at IBM
- Currently working on Samba integration in IBM Spectrum Scale
- Customer support
- Samba team member
- Previous roles:
  - Samba integration and support in other products.
  - Linux device driver maintenance.

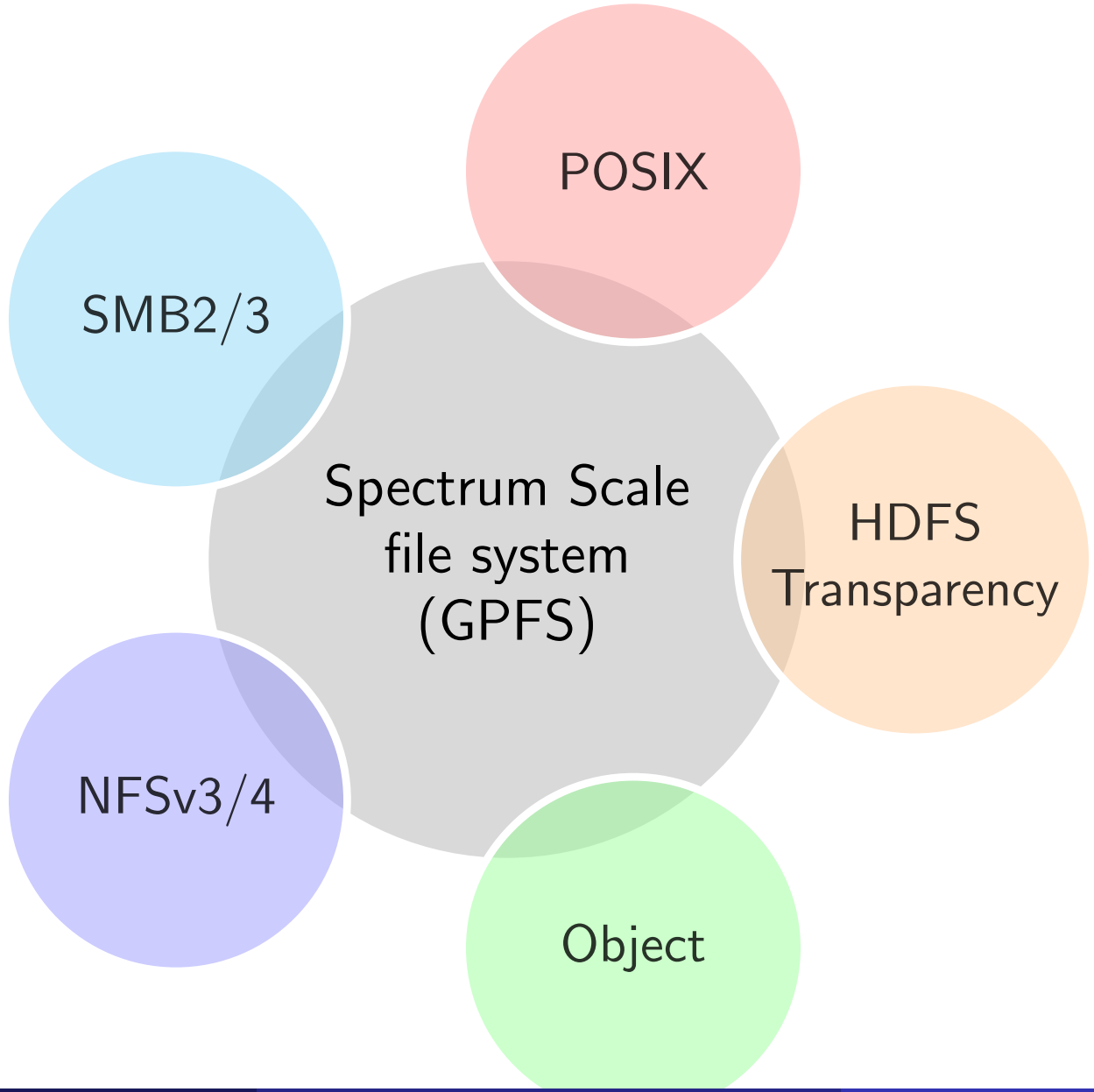
Table of Contents

1. Integrating Samba in IBM Spectrum Scale product
2. Examples of issues faced
3. Lessons learned

Context: IBM Spectrum Scale

- Clustered file system as software defined storage (former name GPFS).
- Include clustered Samba for SMB2 and SMB3 access.
- Support for multiple hardware architectures: x86_64, ppc64le, s390x
- Support for multiple Linux distributions: RHEL, SLES, Ubuntu
- Many other features, but focus here on Samba and SMB.

IBM Spectrum Scale overview

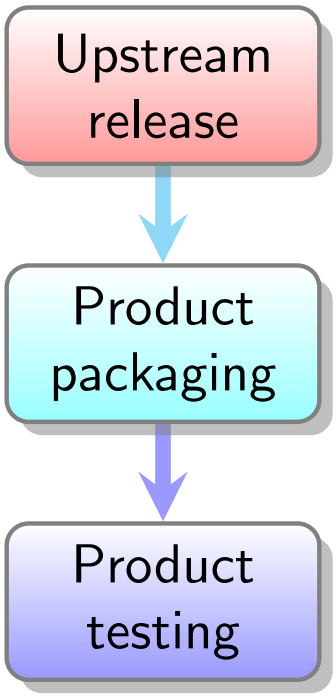


Samba and IBM Spectrum Scale

IBM Spectrum Scale file system (GPFS) has some extensions for SMB:

- DOS attributes as part of metadata
- Create time as part of metadata
- Support for NFSv4 ACLs
- Share mode enforcement in file system
- Cluster-wide file system support for Linux kernel leases
  - For oplocks, not SMB2 leases.
- Implemented in Samba `vfs_gpfs` module that calls file system API.

Packaging Samba for product usage



- Pick a stable upstream release.
- Remove unused code
  - e.g. will never run as Active Directory Domain Controller
- Add additional patches
  - ideally only backports from upstream
- Add config
  - `clustering = yes`
  - `vfs objects = shadow_copy2 syncops gpfs fileid time_audit`
  - `...`
- Package according to product directory structure
  - Avoid conflicts with Linux distribution packages
- Product testing: Functionality, performance, scale-out

Product integration

There are more aspects than packaging the Samba code:

- Service management
  - When to start and stop services
- CLI, GUI
  - Official, supported way of changing config
  - Only allow subset of all Samba options
  - Not feasible to test all combinations of all Samba options
- Floating IP addresses
  - Move to other node in case of node failure
  - Not using ctdb public IP addresses to be independent of ctdb
- Performance counters
- Trace collection, log collection.

Support

- Customer hits a problem, need a way to capture data.
- Capture all logs and config and upload them
- Specifically for Samba
  - level 10 trace
  - network trace (tcpdump)
  - system calls (strace)
- Solution: Provide data capture commands in product CLI

Lesson learned

Data capture needs to be integrated with product and 100% reliable (cannot debug data capture while troubleshooting other problems).

Development view of Samba Integration process

Ideally we would go through this cycle:

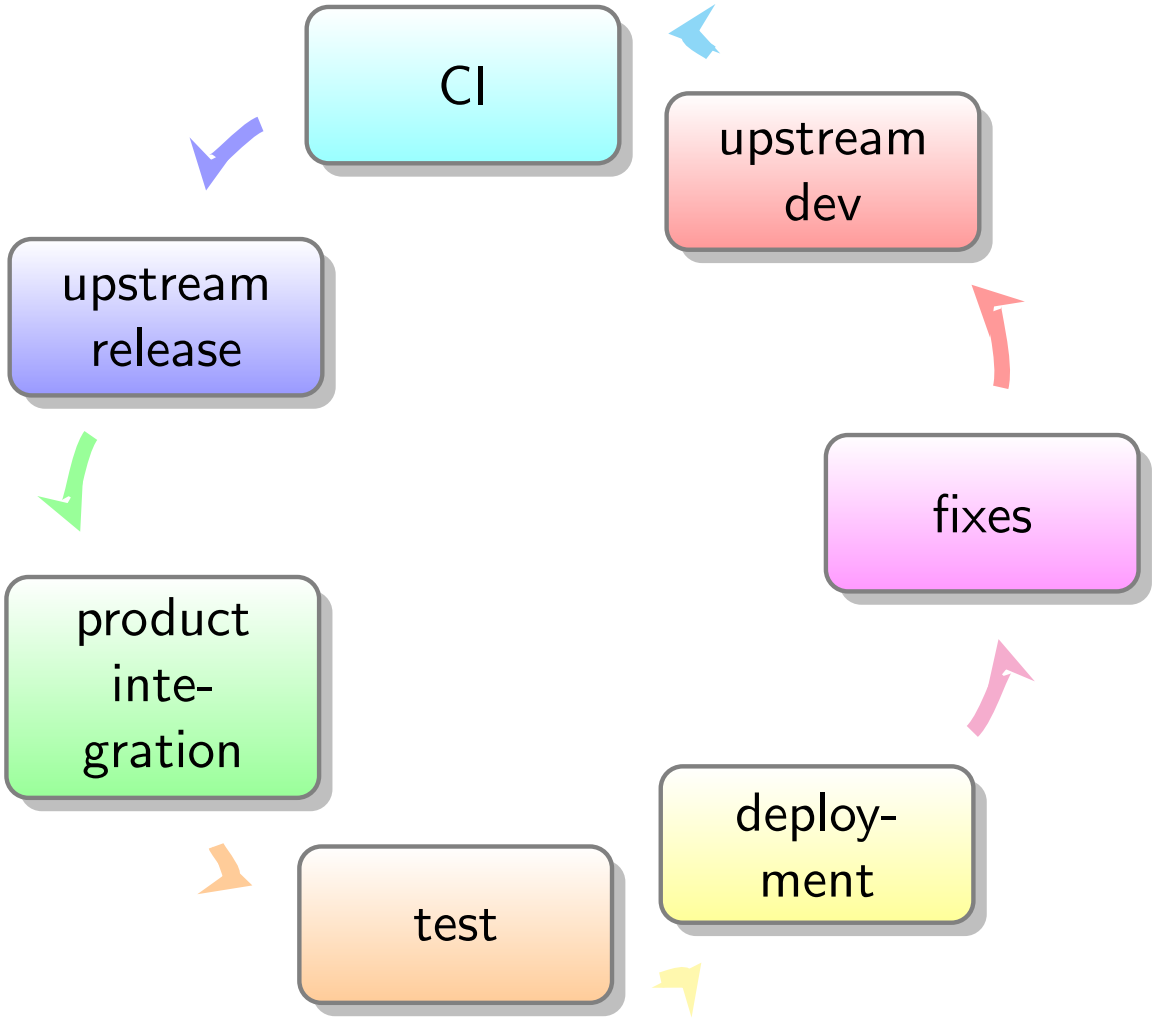


Table of Contents

1 Integrating Samba in IBM Spectrum Scale product

2 Examples of issues faced

3 Lessons learned

Example: Upstream build

- `vfs_gpfs` was not build by default for a long time
- Result: Repeated build breakage over time, had to fix after the fact

```
b620140 vfs_gpfs: Fix compile error in gpfsacl_sys_acl_set_fd
b1b70c1 vfs: Fix the vfs_gpfs build
a06bbf3 vfs_gpfs: Fix compile after change in get_nt_acl_fn
2515ad7 vfs_gpfs: Fix the build with -Werror=declaration-after-statement
c56a88d vfs_gpfs: Fix the build with profiling-data
188d0f0 vfs: Fix compile of vfs_gpfs.c.
83a0b94 s3: Fix the build of vfs_gpfs.c
```

- Luckily only one header file is required for build and that can be redistributed
- Added header file under third_party to build module by default

Lesson learned

Even just building code in upstream CI can prevent breakage

Example: Build with heimdal kerberos library

- Requirement to support Ubuntu
- Debian and Ubuntu use two kerberos libraries (MIT and Heimdal)
- OpenLDAP on Debian and Ubuntu uses Heimdal kerberos
- Samba uses OpenLDAP client libraries
- Do not want to have Samba process use MIT and Heimdal kerberos at the same time.
- Solution: Compile Samba with Heimdal kerberos library
- Problem: Build with system heimdal was broken in Samba 4.8
- Fixed for Samba 4.9, added `--with-system-heimdalkrb5` configure option.
- Done?

Example: Build with heimdal kerberos library

- Samba 4.10 updated Samba build system waf...
- ...which unexpectedly broke the build with system heimdal.
- There was no upstream test.
- Only noticed this particular problem when updating product package to Samba 4.11 and build broke.
- Fixed again:
  `https://bugzilla.samba.org/show_bug.cgi?id=14179`
- Only this time, added also build in upstream CI

Lesson learned

Every important case needs an upstream CI test.

Example: IDMAP_TYPE_BOTH

- Default ID mapping module in our product is `idmap_autorid`
- Fallback if nothing else is configured, also covers BUILTIN and well-known SIDs
- This module returns BOTH, uid and gid for each user and group
- Internally called `IDMAP_TYPE_BOTH`
- Cornercase: File owned by group through "fake" uid (of group)
- `ls -l` will call `getpwuid()` to lookup name
- This worked on Samba 4.5 for "group uid" and broke with Samba 4.6
- Solution: Fix it again and add upstream test: `https://bugzilla.samba.org/show_bug.cgi?id=14141`

Lesson learned

Even corner cases need to be covered in upstream CI testing.

Example: NFSv4 ACL fix

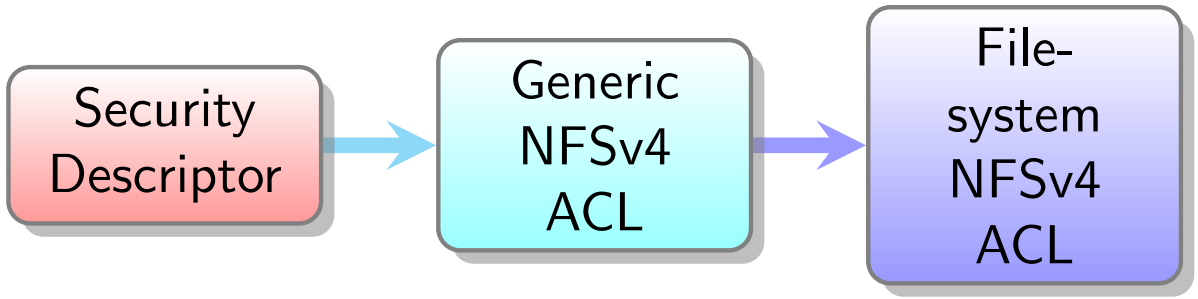
- NFSv4 ACLs are very close to SMB Security Descriptors (SD)
- Inheritance and permission flags map
- Identities need to be mapped from SD SIDs to NFSv4 uids and gids
- Biggest addition in NFSv4 ACLs: “special” entries for owner, group and everyone.
- “special entries” are used by the file system to represent modebits (owner, group, other)
- One source of complexity in the Samba NFSv4 ACL mapping code

Example: NFSv4 ACL fix

- Request to have owner ACL entry mapped to `special:user` in NFSv4 ACL when using IDMAP_TYPE_BOTH
  - Which in turn maps to user modebits
- Seemed obvious enough, changed code
  - `5d4f7bf nfs4acl: Fix owner mapping with ID_TYPE_BOTH`
- Only found out later that this broke the case where file is owned by fake uid for group (to emulate group ownership)
- Why did this get unnoticed?

Example: NFSv4 ACL fix

- Upstream CI has tests that store mapped NFSv4 ACL in xattrs
- No test to validate how mapping between Security Descriptors and NFSv4 ACLs should look like
- Fix this problem and risk breaking another cornercase?
- How to test mapping code which is only used by AIX, GPFS and ZFS?
- NFSv4 ACL mapping path in Samba:



Example: NFSv4 ACL fix

- Solution: Add cmocka unit test to validate mapping between Security Descriptors and generic representation of NFSv4 ACLs
- Then fix problem:
  `https://bugzilla.samba.org/show_bug.cgi?id=14032`
- Could also add unit test for mapping code from generic NFSv4 ACLs to file system NFSv4 ACLs.
- End-to-End testing still needs to happen in special environments as upstream CI only runs Linux with standard file systems.

Lesson learned

When end-to-end testing is not available, unit tests can fill part of the gap.

Example: Hardware accelerated SMB encryption support

- Desire to have SMB encryption using CPU accelerated crypto instructions.
- Nothing available upstream in Samba 4.3 (when this first came up)
- Need support for x86_64 and ppc64le.
- Solution: Change Samba code for product build to use AES functions from openssl
- Test on x86_64, done.

Example: Hardware accelerated SMB encryption support

- Discovered that SMB encryption on ppc64le results in high CPU usage
- Problem: openssl does not always initialize CPU capabilities on ppc64le architecture
- openssl is being used then, but does not use CPU acceleration
- Can be worked around by explicit "init everything" call to openssl
- Need automated test to determine whether CPU is actually used
  - measuring CPU time spent in smbd works reasonable well

Lesson learned

Have automated test for every aspect (here whether CPU encryption acceleration is actually used)

Example: Hardware accelerated SMB encryption support

- Samba 4.12: Upstream Samba now uses GnuTLS.
- Unfortunately no hardware acceleration yet in GnuTLS for POWER: `https://gitlab.com/gnutls/gnutls/-/issues/820`
- Need to keep using openssl for time being to cover POWER architecture

Lesson learned

Deviating from upstream code can cause long-term maintenance burden.

Example: Thread limit

- Testing SLES. Upgrading to newer service pack has smbd crashing under scale-out testing
- Debugging showed that smbd crashed after `pthread_create` returned EGAIN
- Caused by newer systemd having lower nproc ulimit default
- Samba code did not properly handle the error from `pthread_create`
- Fixed: `https://bugzilla.samba.org/show_bug.cgi?id=13170`

Lesson learned

Stress testing on all supported platforms is important, even on minor updates.

Example: Fixes in duplicated quota code

- SMB clients can query free space.
- SMB servers report free space **for user**
- Implemented by querying free disk space and quotas
- `vfs_gpfs` duplicates this logic (historical reason: Samba queried GPFS fileset quota, now gone)
- Now customer wanted to have free space logic consider SGID bit and group quotas
  - SGID on a directory means that files in that directory inherit the primary group from the directory
  - So the interesting quota for free space is now the quota for the primary group on the directory

Example: Fixes in duplicated quota code

- Easy to add the additional check. . . twice. . .

```
vfs_gpfs: Check group quota for directory when SGID is set
quotas: Check group quota for directory when SGID is set
```

Lesson learned

Avoid code duplication. Ideally file system modules like `vfs_gpfs` would only issue API calls and do the minimal data structure mapping for that.

Table of Contents

1 Integrating Samba in IBM Spectrum Scale product

2 Examples of issues faced

3 Lessons learned

Lessons learned

What can be learned from these issues?

- Need to ensure that every aspect is automatically tested in product testing
- Need to test as much as possible in upstream CI as well
  - Avoid breakage during upstream development as much as possible
- Need to pay special attention to features that are not used widely (e.g. NFSv4 ACLs, special file system integration)
- Staying reasonably current with upstream releases, makes applying fixes and handling security issues much easier.
- Fix any problems upstream. Carrying large amount of internal fixes is not sustainable and blocks updates to newer Samba releases.
- Stay away from private code changes, risk of becoming maintenance burden.
- Need stress testing, pushing against resource limits.

Lessons learned

- Clustered Samba is a reliable system.
- Be aware of current limitations.
- Participate in upstream development to integrate with common code as much as possible.

Next steps for `vfs_gpfs`

- Common code should be used for determining free space from quotas.
- Upstream testing: At least a unit test for mapping functions should be doable.
- Expand product test coverage, leverage for code cleanup.
- Long-term: Ideally `vfs_gpfs` module only maps data structures and issues API calls.

Legal Statement

- This work represents the view of the author and does not necessarily represent the view of IBM.
- IBM is a registered trademark of International Business Machines Corporation in the United States and/or other countries.
- Linux is a registered trademark of Linus Torvalds.
- Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Other company, product, and service names may be trademarks or service marks of others.

Questions?