

Shwetha K Acharya

Software Engineer
IBM
Bangalore, India

FOSDEM 2024

Brussels / 3 & 4 February 2024

Exploring Samba on various File Systems:

**Bridging ideas and
enthusiasts together**

What is Samba?

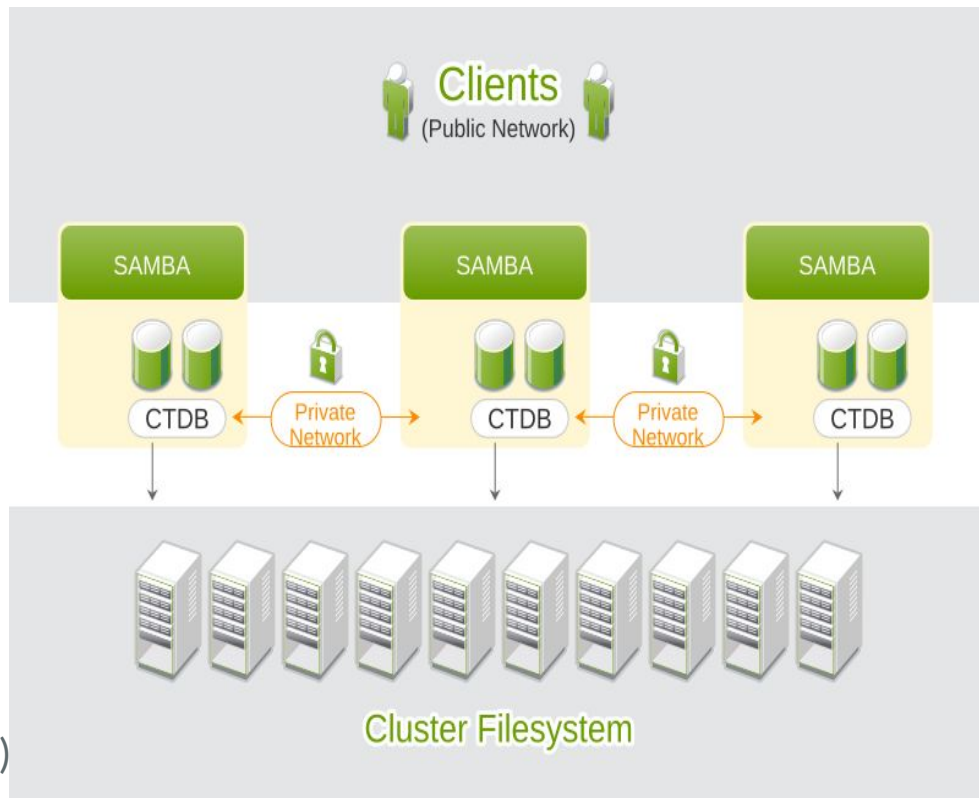
- Let's first understand about SMB and NFS.
- SMB stands for Server Message Block
- It is a protocol used for sharing files, printers, serial ports etc between computers over a network
- In other words I can say that SMB is what NFS is to the UNIX world
- NFS stands for Network File System
- It is also a protocol for allowing user on client computer to access files over a computer network much like a local storage is accessed.

- So now we know that windows and Unix based systems use different protocols for sharing resources over a network.
- But what if we want to use both windows and Unix based systems on a same network and share the same resources among them?
- Long back Dr Andrew Tridgell ended up in a same situation where he wanted to connect his wife's windows PC to his Unix server.
- So he went ahead and wrote a packet sniffer

- As you might be knowing that a packet sniffer would capture and analyze the network traffic.
- It would intercept and log data traveling back and forth in the network, using which one can inspect, monitor and understand the nature of data packets in the network traffic.
- Using the insights he gained from this network sniffer, Dr Tridgel reverse engineered the SMB protocol and implemented it on Unix Box. Thus he made the Unix system appear as a windows server, which allowed him to mount shared file systems from UNIX systems and also concurrently use windows applications.
- He named this project Samba.

How Samba is set up with file system?

- file system is exported on server
- CTDB (Clustered Database) is configured to store temporary data and to ensure High Availability of the server cluster.
- Samba is configured by setting up username, sharename, user password Etc (Samba can also work as domain Controller and respond to security authentication requests like logging in etc)



What is SITE Environment?

- Samba Integration Test Environment is a generic mechanism to setup a clustered Samba network with different file systems.
- Automating such integration environment was initially challenging as it involved many components like cluster nodes and clients, many projects such as Samba, CTDB, GlusterFS/ CephFS/ XFS and also multiple configuration options within these file systems.
- SIT environment solves these challenges by using vagrant-libvirt and ansible.
- Any system that supports above softwares and has minimum resources required by specific file system and its components can get SITE environment up and running. Eg: 2-4 Gib for storage VMs and at least 1 Gib for other VMs.

- You can use this project to automate creation of Samba integration environment on your local machine and experiment things on it, It can also help you debug any issues you face in real time with Samba and any particular file system.
- Apart from that the tests on SIT environment are periodically triggered whenever there is a new change added to the project and also in nightly manner.
- Most recent code from the main will be pulled from the involved projects.

Why SITE ?

- When Samba system exports or uses file system, it allows other computers including those running on different operating systems like microsoft windows/Unix, to access and interact within files in that file system.
- SITE can be configured easily to use different file systems to run tests with different hosts/computers (clients and servers).
- These tests will be very useful to identify:
 - Compatibility issues within Samba and the file system
 - Bugs and unexpected behaviour in the file system
 - Issues in Samba

What are SIT Test Cases?

- This is basically a repository consisting of test suites that run on SIT Environment.
- This suite contains:
 - Consistency tests
 - Container tests
 - Misc tests
 - SMB tests

- Consistency test first mounts the share, creates a new file on it, and unmounts the share. It tests that it can get exactly the same data written on files using different combinations of files and file systems.
- Container tests explore the possibility of running basic tests on a containerized environment
- Misc tests contain tests on basic IO, database IO and stress IO
- Smb tests contain tests from smbtorure framework available in Samba project for exercising various parts of windows protocols and determine how different server implementations respond.

How SITE works?



Host machine



File Edit View Help



Open



Name

▼ QEMU/KVM



playbooks_client0
Running



playbooks_setup0
Running



playbooks_storage0
Running

Custom backends and tests

- Do you want to add your favorite backend/filesystem to SIT environment?
- You will have to:
 - **Define the background** (includes number of machines needed to run the tests, as well as the required resources and roles they perform)
 - **Install dependencies for installation** (includes packages required to install backend components in required machines)
 - **Create main backend role** (includes all the steps required to install, configure, deploy and make the backend accessible on the machine)
 - **Configure CTDB**
 - **Configure Samba**

Refer [docs/backend](#) for detailed information

- Do you want to add custom tests to this project?
Checkout [docs/tests](#)
- Currently tests from [sit-test-cases](#) are run on SITE, you can add custom tests to this project and get them merged.

Reference

- <https://www.samba.org/>
- <https://github.com/samba-in-kubernetes/sit-environment>
- <https://github.com/samba-in-kubernetes/sit-test-cases>
- [https://en.wikipedia.org/wiki/Server Message Block](https://en.wikipedia.org/wiki/Server_Message_Block)
- [https://en.wikipedia.org/wiki/Network File System](https://en.wikipedia.org/wiki/Network_File_System)