DEBUGGING SAMBA-CTDB SETUP IN RED HAT GLUSTER STORAGE

Anoop C S
Software Engineer

Poornima G
Senior Software Engineer

Red Hat, Pune

Friday 22nd June, 2018

AGENDA

- Overview - Samba and SMB
- Configuration and setup
- Clustered Samba with CTDB
- Issue types & Debugging measures
- Performance issues
- Case analysis

SMB and SAMBA

- Samba - server implementation for many Microsoft protocols like SMB, DCERPC etc..
- provides Active Directory and/or Domain Controller services
- SMB file and printer protocol
- implemented in userspace based on documentation available for free
- ubiquitous in nature
- Samba now negotiates and support protocol dialects upto SMB 3.11
- CIFS is usually termed as the successor of SMB
- ports used:
  - 139 – older SMB NetBIOS mode
  - 445 – CIFS

smb.conf

- core configuration file
- mostly located as */etc/samba/smb.conf*
- categorized into different sections
  - [global], [homes], [printers] and shared resources a.k.a shares
- options for variable substitution
- identity mapping options for AD DC services
- man smb.conf(5) explains every parameter pointing to either global (G) or share-specific (S)
- *testparm* for *smb.conf* validation

rhs-samba.conf

- additional configuration file provided with some required parameters set:

  max protocol = SMB3

  kernel share modes = no

  kernel change notify = no

  kernel oplocks = no

  posix locking = no

  store dos attributes = yes

- included via the following line in *smb.conf*:

  include = /etc/samba/rhs-samba.conf

ADDING LOCAL PAM USERS

- *smbpasswd* is your friend

  # smbpasswd -a <local-username>
- user must exist in the system
- user must be added to all Samba servers
- granting permissions to users on share
  - perform a local FUSE mount with *-o acl* option
  - use *setfacl* to set named user permission

    # setfacl -m user:<username>:rwx <mountpoint>
- user restriction using following smb.conf parameters:

  valid users =

  invalid users =

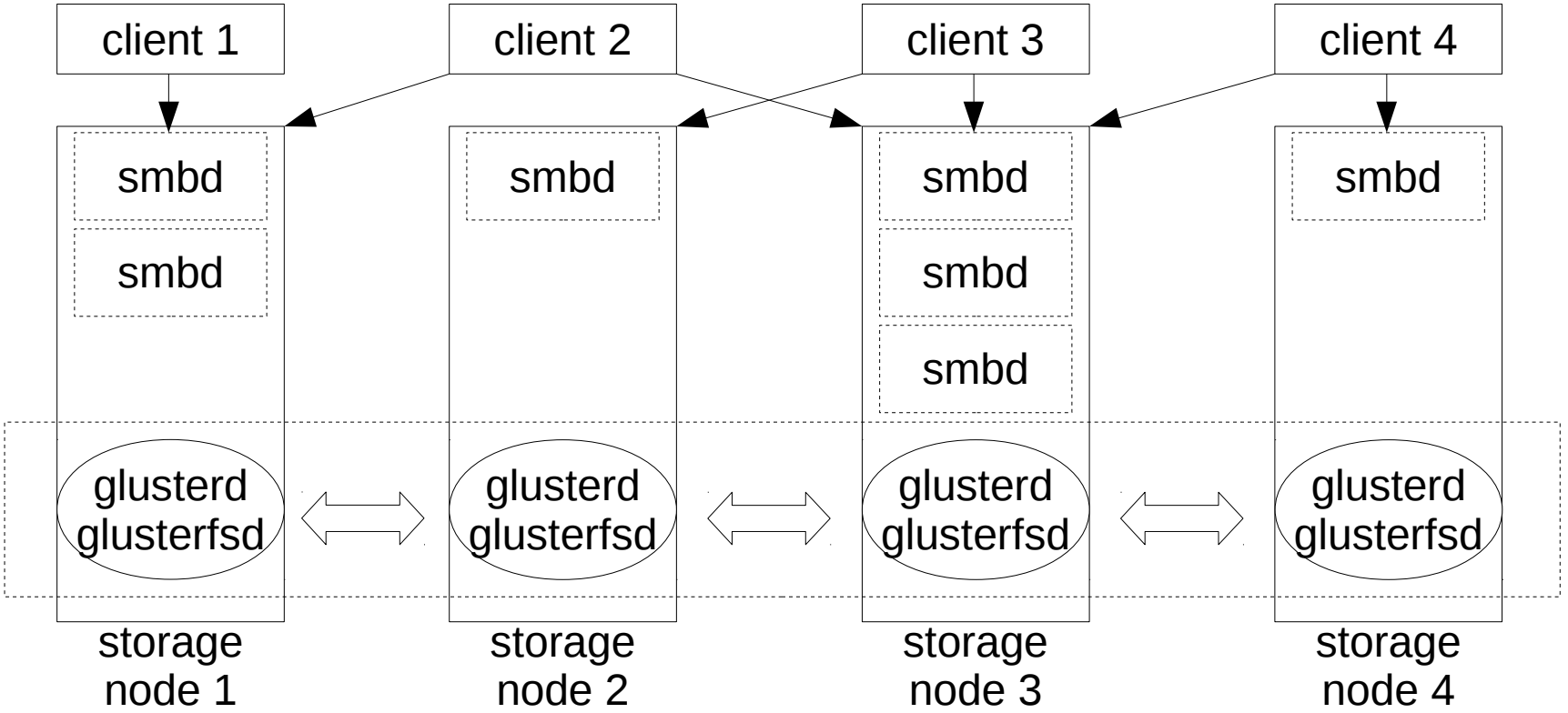
GLUSTERFS SHARE DEFINITION

- supported method via libgfapi
  - libgfapi – native glusterfs API library in C
  - VFS module for glusterfs (vfs_glusterfs)
  - mandated to be the last one in list of vfs modules
  - glusterfs specific options for log level, logfile, volume name etc..
  - automatically added via hook scripts
  - external modifications lost during volume restart
  - each client connection loads entire glusterfs client stack

GLUSTERFS SHARE SECTION

```
[gluster-vol]
    comment = For samba share of volume vol
    guest ok = Yes
    path = /
    read only = No
    vfs objects = glusterfs
    glusterfs:loglevel = 7
    glusterfs:logfile = /var/log/samba/glusterfs-vol.%M.log
    glusterfs:volume = vol
```

STANDALONE SAMBA SERVER SETUP OUTLINE



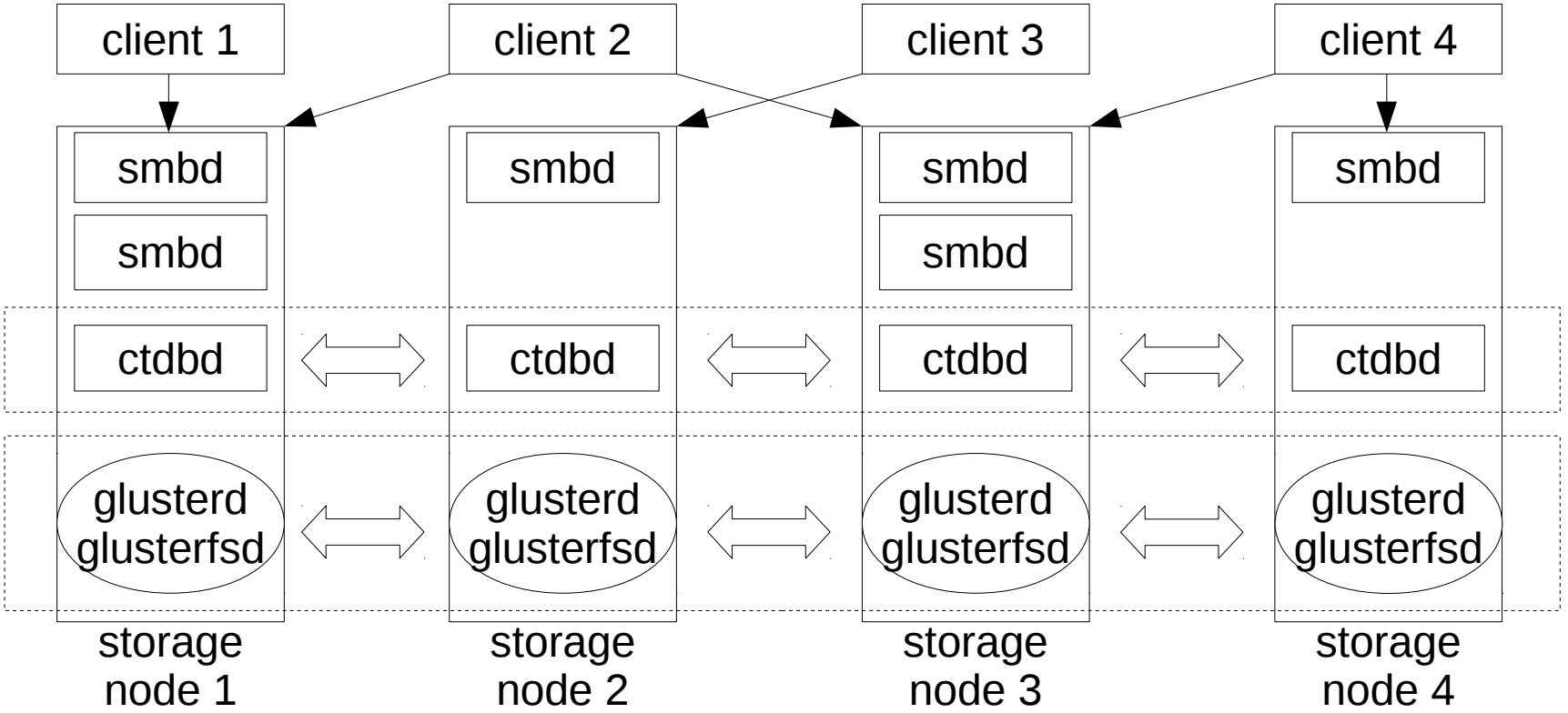
GLUSTERFS SHARE DEFINITION

- unsupported method via FUSE mount re-export
  - used ONLY with approved exception
  - huge number of clients connections to a particular server and/or more volume in use
  - pre-requisite: locally available native glusterfs mount point
  - less memory consumption
  - known issues with Volume Shadow Copy(VSS) service
  - refer k-base article for details

CTDB

- clustered database component in clustered samba for high-availability
- automatically rebuild/recover databases on node failures
- monitor nodes and services
- manage public IP address pool
- load balance services
- private and public addresses in different network
- RECMASTER and LMASTER capabilities

CLUSTERED SAMBA SERVER SETUP OUTLINE



CTDB CONFIGURATION

- various configuration file locations
  - */etc/sysconfig/ctdb*
  - */etc/ctdb/ctdbd.conf*
- contains variables for ctdbd_wrapper script
- commonly used are:
  - CTDB_RECOVERY_LOCK, CTDB_NODES, CTDB_PUBLIC_ADDRESSES, CTDB_MANAGES_SAMBA, CTDB_MANAGES_WINBIND, CTDB_LOGGING
- associated with smb.conf clustering parameter

CTDB RECOVERY LOCK FILE

- split brain avoidance
- election of recovery master
- need for separate volume to keep lock file
- native glusterfs mount of CTDB lock volume on every node
- */etc/fstab* entry for automatic mount (preferably with *x-systemd.requires=glusterd*)
- an improvement to ctdb systemd service file
  - Adding *network.target*
- preferred to be inside a directory under root of the volume
- CTDB lock is not recommended to be shared via Samba

SMB/CIFS CLIENT SCOPE

- kernel CIFS module
- *smbclient* utility from Samba project
- windows
- mac OS X

ISSUE TYPES

- Package update/install problems
- Unhealthy Samba-CTDB cluster
- Functional anomalies
- Process crashes
- Slow performance
- Null/Erroneous/enormous log entries
- Memory consumption/leaks
- Client driven

TROUBLESHOOTING RPM/PACKAGING ISSUES

- whether reproducible or not?
  - Follow the steps from customer setup
- verify the list of packages installed
  - Is the conflict caused due to some unwanted package?
  - Any pre-requisite packages are missing?
- check the source(channel) for installed packages
- apart from glusterfs packages, following should be present:
  - libtdb, libtalloc, libtevent and libldb from gluster samba channel
  - samba and derivatives(samba-libs, samba-common etc), samba-vfs-glusterfs from gluster samba channel

TROUBLESHOOTING RPM/PACKAGING ISSUES

- useful commands:
  - *yum list <package-name>*
    lists the available package with repo name
  - *yum whatprovides "*/<binary or file>"*
    lists the package which contains the specified binary/file
  - *rpm -qf <path to file or binary>*
    lists the package with which the specified file/binary was installed
  - *yum repolist*
    lists the enabled repos in the system

UNHEALTHY SAMBA CLUSTER

- is CTDB lock mounted?

  # mount -a

  # mount | grep <ctdb-lock-volume-name>
- if mounted, check whether it is accessible

  # ls -al <mount-point> or try to touch a file
- look for abnormal entries in */var/log/log.ctdb*
  - *ERROR: Daemon able to take recovery lock on "/gluster/lock/lockfile" during recovery*
  - *Unable to take recovery lock – contention*
  - *50.samba: Failed to start smb.service: Access denied*

CTDB NODE STATUS

- CTDB node can be in one of the following states:
  - OK – all fine
  - DISCONNECTED – not reachable via private network
  - DISABLED – disabled to not host public addresses for services
  - UNHEALTHY – failed a health check
  - BANNED – not behaving as designed
  - STOPPED – administratively excluded from cluster
  - PARTIALLYONLINE – participates like a healthy nodes with down ifaces

GENERAL SAMBA FUNCTIONAL ISSUES

- verify smb.conf from sosreport using *testparm*
  - *testparm -s /etc/samba/smb.conf*
- try to collect the reproducer steps
- presence of errors from any of the following log files:
  - */var/*log/samba/log.
  - */var/*log/samba/log.smbd
  - */var/*log/samba/log.<client-ip>
  - */var/*log/samba/log.<client-hostname>
- check for entries with ‘E’ in glusterfs client log under */var/log/samba/glusterfs-<volanme>.<ip>.log*

GENERAL SAMBA FUNCTIONAL ISSUES

- also check for coredumps
  - coredump file location determined by */proc/sys/kernel/core_pattern*
  - *backtrace in /var/log/samba/log.** and/or in */var/log/messages*
- increasing logs for debugging:
  - set the following parameters in *smb.conf*

    log level = 10, *debug pid = true andmax log size = 0*

    verify changes by running *testparm*, clear current logs and restart services
  - or to avoid interrupt in services

    # smbcontrol <smbd/nmbd/winbindd> debug 10
- *smbstatus*

SMBD/CTDBD CRASH REPORT

- backtrace presence in */var/log/messages* or */var/log/log.**
- coredumps are generally handled by *abrt-ccpp* service using *abrt-hook-ccp* binary

  /usr/libexec/abrt-hook-ccpp %s %c %p %u %g %t e %P %I %h
- identify which process crashed

  # file <path-to-coredump>
- for smbd crashes it can be either samba or glusterfs issue
  - if backtrace suggests /usr/lib64/samba/vfs/glusterfs.so it can very well be a glusterfs problem
  - even otherwise glusterfs can be involved

PERFORMANCE PROBLEMS

- validate *smb.conf*
- check for following glusterfs volume set options based on workload and setup:
  - Meta-data cache

    features.cache-invalidation, *features.cache-invalidation-timeout*, *performance.stat-prefetch*, *performance.cache-samba-metadata*, *performance.cache-invalidation*, *performance.md-cache-timeout*
  - Negative lookup cache

    features.cache-invalidation, *features.cache-invalidation-timeout*, *performance.nl-cache*
  - Parallel readdir

    performance.parallel-readdir

PERFORMANCE PROBLEMS

- try to understand the access pattern
- network traces from client and/or server
  - use Wireshark for Windows, tcpdump for Linux client/servers

    # tcpdump -i any -s0 -w /tmp/samba.pcap host <ip>
  - Make sure that the capture contains SMB requests/replies while issue was encountered
- profile information from glusterfs
  - Clear any current profile info – start profiling using *gluster* cli – Perform reproducer steps
  - Stop profiling
  - Output the profile info

UNDESIRED LOG ENTRIES

- how frequent?
- when are they being logged?
- what is the functional impact?
- are they being logged as errors?
- no log entries at all?
  - */var/*log/samba/log.<ip>
  - *# ls -lrt /var/log/samba/*

    lists different log files based on how recently they were updated

MEMORY CONSUMPTION

- memory consumption in idle situation
  - check for any stale smbd process from *smbstatus*
  - get process stack using *pstack*

    # pstack <pid>
  - are there any client side heal in process?
- memory consumption stays at higher value after an active IO
  - chance of memory leak
  - request and compare statedumps before and after IO

MEMORY CONSUMPTION

- Possible constituents
  - number of different clients
  - number of different volumes
- How much memory?
  - use *top* command line utility
  - best case - ~550 MB VIRT, 11 MB RSS and can only increase
  - can vary across systems and under different workloads
  - check if either VIRT or RSS increases gradually
  - number of different clients connected to/accessing shares
  - number of different glusterfs volumes

CLIENT ISSUES

- For any issue
  - is it reproducible via other means within same client? explorer, finder, powershell, network mapped drive, xcopy. . .
  - try with smbclient, if reproducible get more debug info with *-d10*
  - may be with kernel CIFS client too
  - network trace from client side
  - consider smbclient as your next best SMB client
  - */var/log/messages* for kernel CIFS client errors

CASE ANALYSIS

- slowness in copying small files from windows 7 client [bug]
- CTDB in UNHEALTHY state [bug]
- Milestone(CCTV footage management software) certification
- regular smbd crashes from libgfapi [bug]
- Package conflicts [bug]
- SELinux issues

Thank you

@anoopcs
anoopcs@redhat.com

@poornimag
pgurusid@redhat.com

Q & A