RED HAT GLUSTER STORAGE ADVANCED FEATURES LAB

Dustin L. Black, Principal Technical Account Manager
Guil Barros, Principal Product Manager

June 25, 2015

Dustin L. Black, RHCA
@dustinlblack
dustin@redhat.com

Guil Barros, RHCA
@gfb
gbarros@redhat.com

#rhsummit #redhat #gluster #fearthebeard

redhat

http://goo.gl/eQcuVP

redhat

In this lab you will...

- Explore the GlusterFS data layout and extended attributes
- Understand the structure of the volfiles
- Administer quotas on volumes and subdirectories
- Induce a data split brain and observe the effects
- Administer quorum and understand its actions
- Configure asynchronous geo-replication
- Take snapshots, observe the operations, and restore data
- Configure and understand disperse volumes (erasure coding)

Know your lab

root password: **redhat**

Trusted Pool #1

- Nodes: n1 - n4
- IPs: 10.11.12.101 - 104
- Volume: rep01 - Dist-Rep 2x2

Trusted Pool #2

- Nodes: n5 - n6
- IPs: 10.11.12.105 - 106
- Volume: srep01 - Dist

Disclaimer:

This is alpha-level code we're working with in the lab. Here be dragons.

The Magic .glusterfs Directory

- Visible from the brick
- File metadata is stored with the files in extended attributes
  - **trusted.gfid** value for each file
- Hardlink file in a special directory structure based on gfid

```
trusted.gfid = 0x57ab4dd8afae41eda54f6ecba5985a6b
Hardlink = <brick>/.glusterfs/57/ab/57ab4dd8afae41eda54f6ecba5985a
6b
```

The Magic .glusterfs Directory

```
[root@n1 ~]# getfattr -d -m . -e hex /rhgs/bricks/rep01/file002
getfattr: Removing leading '/' from absolute path names
# file: rhgs/bricks/rep01/file002
trusted.afr.dirty=0x000000000000000000000000
trusted.afr.rep01-client-0=0x000000000000000000000000
trusted.afr.rep01-client-1=0x000000000000000000000000
trusted.bit-rot.version=0x02000000000000000556cac8f0008d758
trusted.gfid=0x57ab4dd8afae41eda54f6ecba5985a6b
```

The Magic .glusterfs Directory

```
[root@n1 ~]# find /rhgs/bricks/rep01 -samefile /rhgs/bricks/rep01/\
.glusterfs/57/ab/57ab4dd8-afae-41ed-a54f-6ecba5985a6b | xargs ls -li
136 -rw-r--r-- 2 root root 1048576 Jun  1 15:08 /rhgs/bricks/rep01/file002
136 -rw-r--r-- 2 root root 1048576 Jun  1 15:08
/rhgs/bricks/rep01/.glusterfs/57/ab/57ab4dd8-afae-41ed-a54f-6ecba5985a6b
```

Volfiles

- Volume functionality is layered in a hierarchy
- Config file on the server ; In-memory on the client
- Brick volumes are defined first
- Replica volumes contain brick subvolumes
- DHT volume contains replica or brick subvolumes
- Additional functionality layered on the DHT volume

Volfiles

```
[root@n1 ~]# cat /var/lib/glusterd/vols/rep01/rep01.tcp-fuse.vol
volume rep01-client-0
...
    option remote-subvolume /rhgs/bricks/rep01
    option remote-host n1...
end-volume

...

volume rep01-replicate-0
    type cluster/replicate
    subvolumes rep01-client-0 rep01-client-1
end-volume

...

volume rep01-dht
    type cluster/distribute
    subvolumes rep01-replicate-0 rep01-replicate-1
end-volume
```

Volfiles

Don't forget to reset your volume options before continuing

```
[root@n1 ~]# gluster volume reset rep01 force
volume reset: success: reset volume successful
```

Quotas

- Per directory ; Not per-user
- Set at the directory level
- Subdirs must be *client-created* before a quota can be applied
- **quota-deem-statfs** feature needed for client mount point to reflect the applied quota

Quotas

```
[root@n1 ~]# gluster volume quota rep01 limit-usage / 200MB
volume quota : success
[root@n1 ~]# gluster volume quota rep01 limit-usage /testdir 50MB
volume quota : success
[root@n1 ~]# gluster volume quota rep01 list
                  Path                   Hard-limit Soft-limit   Used
Available  Soft-limit exceeded? Hard-limit exceeded?
-----------------------------------------------------------------------
-------------------------------------------
/                                         200.0MB       80%     101.0MB
 99.0MB              No                    No
/testdir                                   50.0MB       80%      0Bytes
 50.0MB              No                    No
```

redhat

Quotas

```
[root@n1 ~]# df -h /rhgs/client/rep01/
Filesystem      Size  Used Avail Use% Mounted on
n1:rep01         16G  167M   16G   2% /rhgs/client/rep01
[root@n1 ~]# gluster volume set rep01 features.quota-deem-statfs on
volume set: success
[root@n1 ~]# df -h /rhgs/client/rep01/
Filesystem      Size  Used Avail Use% Mounted on
n1:rep01        200M  101M   99M  51% /rhgs/client/rep01
```

Quotas

Don't forget to reset your volume options before continuing

```
[root@n1 ~]# gluster volume reset rep01 force
volume reset: success: reset volume successful
```

Split-Brain

- Triggered by network or data path interruptions
- Different commits succeed at different bricks
- Cannot be automatically resolved
  - Which copy is the *good* copy? It might be both...
- File metadata and logs are key to troubleshooting

Split-Brain

```
[root@n1 ~]# getfattr -d -m . -e hex /rhgs/bricks/rep01/file002 \
 | grep afr.rep01
getfattr: Removing leading '/' from absolute path names
trusted.afr.rep01-client-0=0x000000000000000000000000
trusted.afr.rep01-client-1=0x0000003a0000000000000000
```

```
[root@n2 ~]# getfattr -d -m . -e hex /rhgs/bricks/rep01/file002 \
 | grep afr.rep01
getfattr: Removing leading '/' from absolute path names
trusted.afr.rep01-client-0=0x0000001e0000000000000000
trusted.afr.rep01-client-1=0x000000000000000000000000
```

```
0x 000003d7 00000001 00000110
       |      |       |
       |      |       \_ changelog of directory entries
       |      \_ changelog of metadata
       \ _ changelog of data
```

Split-Brain

```
[root@n1 ~]# gluster volume heal rep01 info split-brain
Brick n1:/rhgs/bricks/rep01/
/file002
Number of entries in split-brain: 1

Brick n2:/rhgs/bricks/rep01/
/file002
Number of entries in split-brain: 1
...
```

```
[root@n1 ~]# grep split /var/log/glusterfs/rhgs-client-rep01.log
[2015-06-11 22:21:55.757038] W [MSGID: 108008] [afr-read-txn.c:241:afr_read
_txn] 0-rep01-replicate-0: Unreadable subvolume -1 found with event generat
ion 4. (Possible split-brain)
```

Split-Brain

```
[root@n2 ~]# setfattr -n trusted.afr.rep01-client-0 \
 -v 0x000000000000000000000000 /rhgs/bricks/rep01/file002
[root@n2 ~]# getfattr -d -m . -e hex /rhgs/bricks/rep01/file002 \
 | grep afr.rep01
getfattr: Removing leading '/' from absolute path names
trusted.afr.rep01-client-0=0x000000000000000000000000
trusted.afr.rep01-client-1=0x000000000000000000000000
```

Split-Brain

Don't forget to reset the split files before continuing

```
[root@n1 ~]# ~/split_reset.sh
Flushing firewall...
$ iptables -F
Deleting split files...
...
```

Quorum - Server-Side

- Pool-aware only
  - Looks for >50% availability of trusted pool nodes
- Data consistency enforced by killing bricks
  - Abrupt and complete loss of data access

```
[root@n1 ~]# gluster volume set rep01 cluster.server-quorum-type server
volume set: success
```

Quorum - Server-Side

Don't forget to reset the split files before continuing

```
[root@n1 ~]# ~/split_reset.sh
Flushing firewall...
$ iptables -F
Deleting split files...
...
```

Quorum - Server-Side

```
[root@n1 ~]# grep quorum \
 /var/log/glusterfs/etc-glusterfs-glusterd.vol.log
…
[2015-06-08 15:51:57.558778] C [MSGID: 106002] [glusterd-server-quorum
.c:356:glusterd_do_volume_quorum_action] 0-management: Server quorum l
ost for volume rep01. Stopping local bricks.
```

```
[root@n1 ~]# grep quorum \
 /var/log/glusterfs/etc-glusterfs-glusterd.vol.log
…
[2015-06-08 16:05:02.430800] C [MSGID: 106003] [glusterd-server-quorum
.c:351:glusterd_do_volume_quorum_action] 0-management: Server quorum r
egained for volume rep01. Starting local bricks.
```

Quorum - Server-Side

Don't forget to reset your volume options before continuing

```
[root@n1 ~]# gluster volume reset rep01 force
volume reset: success: reset volume successful
```

Quorum - Client-Side

- Volume- and replica-aware
  - Looks for >50% availability of a replica set
  - An extra *vote* is given to the first brick by default
- Data consistency enforced by setting non-quorate bricks read-only
  - All data remains readable ; write access is degraded

```
[root@n1 ~]# gluster volume set rep01 cluster.quorum-type auto
volume set: success
```

Quorum - Client-Side

```
[root@n2 ~]# grep quorum /var/log/glusterfs/rhgs-client-rep01.log
…
[2015-06-08 16:35:06.112351] W [MSGID: 108001] [afr-common.c:3963:afr_
notify] 0-rep01-replicate-0: Client-quorum is not met
```

```
[root@n2 ~]# grep quorum /var/log/glusterfs/rhgs-client-rep01.log
…
[2015-06-08 16:50:50.802194] I [MSGID: 108002] [afr-common.c:3959:afr_
notify] 0-rep01-replicate-0: Client-quorum is met
```

Geo-Replication

- Asynchronous one-way replicate to a remote volume
- Active and passive members of each replica set
  - Active members copy data in parallel
- An initial crawl scans the source volume for changes
- Ongoing changes are recorded in a changelog to avoid further scanning overhead
- Changes are batched and sent to an rsync or tar+ssh daemon

Geo-Replication

```
[root@n1 ~]# gluster volume geo-replication rep01 n5::srep01 create push-pem
Creating geo-replication session between rep01 & n5::srep01 has been successful
[root@n1 ~]# gluster volume geo-replication rep01 status

MASTER NODE    MASTER VOL    MASTER BRICK          SLAVE USER    SLAVE
 SLAVE NODE    STATUS     CRAWL STATUS    LAST_SYNCED
------------------------------------------------------------------------------
------------------------------------------------
n1             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Created    N/A             N/A
n4             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Created    N/A             N/A
n3             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Created    N/A             N/A
n2             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Created    N/A             N/A
```

redhat

Geo-Replication

```
[root@n1 ~]# gluster volume geo-replication rep01 n5::srep01 start &&
  watch -n .5 gluster volume geo-replication rep01 n5::srep01 status
Starting geo-replication session between rep01 & n5::srep01 has been successful

MASTER NODE    MASTER VOL    MASTER BRICK          SLAVE USER    SLAVE
 SLAVE NODE    STATUS             CRAWL STATUS    LAST_SYNCED
-------------------------------------------------------------------------------
-----------------------------------------------------------
n1             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Initializing...    N/A             N/A
n2             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Initializing...    N/A             N/A
n4             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Initializing...    N/A             N/A
n3             rep01         /rhgs/bricks/rep01    root          ssh://n5::srep01
 N/A           Initializing...    N/A             N/A
```

redhat

Geo-Replication

Don't forget to stop geo-replication before continuing

```
[root@n1 ~]# gluster volume geo-replication rep01 n5::srep01 stop
Stopping geo-replication session between rep01 & n5::srep01 has b
een successful
```

Snapshots

- Point-in-time volume copies
- Copy-on-write using LVM thin provisioning
  - Near-instantaneous with minimal overhead
- Orchestrates LVM snapshots across all bricks
  - LVM thin pool configuration is prerequisite

Snapshots

```
[root@n1 ~]# gluster snapshot create snap01 rep01
snapshot create: success: Snap snap01_GMT-2015.06.09-19.08.34
created successfully
```

```
[root@n1 ~]# lvs
  LV                                   VG      Attr       LSize   Pool
 Origin  Data%  Meta%  Move Log Cpy%Sync Convert
  root                                 rhel    -wi-ao----   4.40g
  swap                                 rhel    -wi-ao---- 512.00m
  a9aa400404a8477b89823c51bbebeb91_0 rhgs_vg Vwi-aot---   8.00g rhgs_pool
 rhgs_lv 1.21
  rhgs_lv                              rhgs_vg Vwi-aot---   8.00g rhgs_pool
         1.21
  rhgs_pool                            rhgs_vg twi-aot---   8.00g
         1.24   0.02
[root@n1 ~]# df -h | grep snaps
/dev/mapper/rhgs_vg-a9aa400404a8477b89823c51bbebeb91_0  8.0G  118M  7.9G
2% /run/gluster/snaps/a9aa400404a8477b89823c51bbebeb91/brick1
```

Snapshots

```
[root@n1 ~]# snap01=`gluster snapshot list rep01 | grep snap01`
[root@n1 ~]# gluster snapshot info $snap01
Snapshot                  : snap01_GMT-2015.06.11-23.06.26
Snap UUID                 : 86716030-c258-4643-9e22-53b646e62d6c
Created                   : 2015-06-11 23:06:26
Snap Volumes:

        Snap Volume Name          : a9aa400404a8477b89823c51bbebeb91
        Origin Volume name        : rep01
        Snaps taken for rep01      : 1
        Snaps available for rep01  : 255
        Status                    : Stopped
```

Snapshots

Activating the snapshot, we can now mount it to the client as read-only

```
[root@n1 ~]# snap02=`gluster snapshot list rep01 | grep snap02`
[root@n1 ~]# gluster snapshot activate $snap02
Snapshot activate: snap02_GMT-2015.06.09-19.46.17: Snap activated successfully
[root@n1 ~]# mkdir /rhgs/client/snap02
[root@n1 ~]# mount -t glusterfs n1:/snaps/$snap02/rep01 /rhgs/client/snap02
[root@n1 ~]# mount | grep snap02
n1:/snaps/snap02_GMT-2015.06.09-19.46.17/rep01 on /rhgs/client/snap02 type fuse.gl
terfs (ro,relatime,user_id=0,group_id=0,default_permissions,allow_other,max_read=1
072)
[root@n1 ~]# ls /rhgs/client/snap02/newfile001
/rhgs/client/snap02/newfile001
```

Snapshots

The User-Serviceable Snapshots feature allows the user to directly access any activated snapshots

```
[root@n1 ~]# gluster volume set rep01 features.uss enable
volume set: success
[root@n1 ~]# ls /rhgs/client/rep01/.snaps/$snap02/newfile001
/rhgs/client/rep01/.snaps/snap02_GMT-2015.06.09-19.46.17/newfile001
```

Disperse Volumes (Erasure Coding)

- Data protection through dispersed redundancy
- Configurable redundancy ; Parity similar to RAID 5/6
- More efficient storage use vs. replication
- Higher levels of data protection available
- Remove dependence on underlying RAID
- Tradeoffs
  - Higher overhead due to parity calculations
  - Data is broken up among bricks and is not accessible offline

Disperse Volumes (Erasure Coding)

We define the total number of bricks, and the number of brick failures we can tolerate

```
[root@n1 ~]# gluster volume create ec01 disperse 6 redundancy 2 \
 n1:/rhgs/bricks/ec01-1 n2:/rhgs/bricks/ec01-1 n3:/rhgs/bricks/ec01-1 \
 n4:/rhgs/bricks/ec01-1 n1:/rhgs/bricks/ec01-2 n2:/rhgs/bricks/ec01-2 force
```

Disperse Volumes (Erasure Coding)

```
[root@n1 ~]# gluster volume info ec01

Volume Name: ec01
Type: Disperse
Volume ID: f9f8d1d8-10d0-48cf-8292-a03860296b80
Status: Started
Number of Bricks: 1 x (4 + 2) = 6
Transport-type: tcp
Bricks:
Brick1: n1:/rhgs/bricks/ec01-1
Brick2: n2:/rhgs/bricks/ec01-1
Brick3: n3:/rhgs/bricks/ec01-1
Brick4: n4:/rhgs/bricks/ec01-1
Brick5: n1:/rhgs/bricks/ec01-2
Brick6: n2:/rhgs/bricks/ec01-2
Options Reconfigured:
performance.readdir-ahead: on
```

Disperse Volumes (Erasure Coding)

```
[root@n1 ~]# cat /var/lib/glusterd/vols/ec01/ec01.tcp-fuse.vol
...
volume ec01-disperse-0
    type cluster/disperse
    option redundancy 2
    subvolumes ec01-client-0 ec01-client-1 ec01-client-2 ec01-client-3 ec01-c
lient-4 ec01-client-5
end-volume
...
```

Your feedback is appreciated, and will help us continue to improve

Please complete the session survey

http://goo.gl/eQcuVP

redhat

@dustinlblack @gfb

#rhsummit #redhat #gluster #fearthebeard