

[搜索](#)[正文](#)[标题](#)

GlusterFS Storage Cluster on CentOS 7

Contents

1. Introduction
2. Terms
3. Environment
4. **GlusterFS** Server Installation
 1. XFS Bricks
 2. Trusted Pool (Storage Cluster)
5. High Availability **GlusterFS** Volumes
6. **GlusterFS** Clients
 1. Open the Firewall for **Glusterfs**/NFS/CIFS Clients
 2. Access from another CentOS 7 machine via **GlusterFS** Native Client
 3. Access from other Linux machines via NFS
 4. Troubleshooting NFS
 5. Access from Windows/Linux machines via CIFS
 6. Troubleshooting SELinux issues
7. Extend **GlusterFS** Volumes without downtime

1. Introduction

This article describes the deployment steps of a high availability **GlusterFS** Storage environment on CentOS 7.

After the basic concepts, this document provides information about the following:

- Gluster Architecture Installation
- Create Highly Available (Replicated) **GlusterFS** Volumes
- Connect from Linux/Windows clients
- Extend **GlusterFS** Volumes without downtime

Reference System:

```
# hostnamectl
  Static hostname: gluster1.example.com
        Icon name: computer
        Chassis: n/a
        Machine ID: b62adea1c2ca472ab04bccafea769109
        Boot ID: c315fd81d1884de4bbf74209caf41c51
  Virtualization: kvm
  Operating System: CentOS Linux 7 (Core)
        CPE OS Name: cpe:/o:centos:centos:7
        Kernel: Linux 3.10.0-229.11.1.el7.x86_64
  Architecture: x86_64
```

GlusterFS Packages:

```
# rpm -qa |grep gluster
samba-vfs-glusterfs-4.1.12-23.el7_1.x86_64
glusterfs-server-3.7.3-1.el7.x86_64
glusterfs-libs-3.7.3-1.el7.x86_64
glusterfs-client-xlators-3.7.3-1.el7.x86_64
glusterfs-api-3.7.3-1.el7.x86_64
glusterfs-fuse-3.7.3-1.el7.x86_64
glusterfs-3.7.3-1.el7.x86_64
glusterfs-cli-3.7.3-1.el7.x86_64
```

2. Terms

- Gluster Storage server: The machine which hosts the file system in which data will be stored.
- Gluster Storage client: The machine which mounts the **GlusterFS** Volume (this may also be a server).
- Brick: The brick is a disk partition with XFS file system that has been assigned to a Volume.
- **GlusterFS** Volume: The logical collection of bricks.

3. Environment

The basic installation will be demonstrated on two nodes act as a storage cluster.

Servers	gluster1.example.com	gluster2.example.com
	CentOS 7.1 x64	CentOS 7.1 x64
Clients	centos7.example.com	win2k8srv.example.com
	CentOS 7 x64	Windows Server 2008 x64

SELinux operates in enforcing mode and the firewall is enabled.

```
# sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:      /etc/selinux
Loaded policy name:           targeted
Current mode:                 enforcing
Mode from config file:       enforcing
Policy MLS status:           enabled
Policy deny_unknown status:   allowed
Max kernel policy version:    28
```

```
# firewall-cmd --state
running
```

Both servers have an empty disk attached as `/dev/vdb`. Using this disk, a new logical volume group

(vg_gluster) will be created for XFS bricks.

Verify available partitions:

```
# cat /proc/partitions
major minor #blocks name
252      0    31457280 vda
252      1     512000 vda1
252      2    30944256 vda2
252     16    31457280 vdb
... output omitted ...
```

4. GlusterFS Server Installation

```
# yum update -y
```

Download the latest *glusterfs-epel* repository from *gluster.org*:

```
# yum install wget -y

# wget -P /etc/yum.repos.d/ http://download.gluster.org/pub/gluster
/glusterfs/LATEST/CentOS/glusterfs-epel.repo
```

Also install the latest *EPEL* repository from *fedoraproject.org* to resolve all dependencies:

```
# yum install http://dl.fedoraproject.org/pub/epel/epel-release-
latest-7.noarch.rpm
```


Both repositories are enabled by default:

```
# yum repolist
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* base: repo.bigstepcloud.com
* epel: epel.check-update.co.uk
* extras: centos.serverspace.co.uk
* updates: mirrors.coreix.net
repo id          repo
name
status
base/7/x86_64    CentOS-7 -
Base
8,652
epel/x86_64      Extra Packages for Enterprise Linux
7 - x86_64      8,476
extras/7/x86_64  CentOS-7 -
Extras
180
glusterfs-epel/7/x86_64    GlusterFS is a clustered
file-system capable of scaling to several petabytes.      14
glusterfs-noarch-epel/7    GlusterFS is a clustered
file-system capable of scaling to several petabytes.      2
updates/7/x86_64  CentOS-7 -
Updates
1,246
repolist: 18,570
```

Install **GlusterFS** Server and Samba packages on both storage cluster nodes.

```
# yum install glusterfs-server samba -y
```

4.1. XFS Bricks

 Skip these optional steps if your environment is already have prepared XFS bricks (partitions).

Create a new *Physical Volume* using /dev/vdb disk:

```
# pvcreate /dev/vdb
Physical volume "/dev/vdb" successfully created
```

Create a *Volume Group* on /dev/vdb:

```
# vgcreate vg_gluster /dev/vdb
Volume group "vg_gluster" successfully created
```

Create **brick1** and **brick2** *Logical Volumes* for XFS bricks on both cluster nodes:

```
# lvcreate -L 5G -n brick1 vg_gluster
Logical volume "brick1" created.

# lvcreate -L 5G -n brick2 vg_gluster
Logical volume "brick2" created.
```

Setup *XFS* file systems:

```
# mkfs.xfs /dev/vg_gluster/brick1
# mkfs.xfs /dev/vg_gluster/brick2
```

Create mount points and mount XFS bricks:

```
# mkdir -p /bricks/brick{1,2}
# mount /dev/vg_gluster/brick1 /bricks/brick1
# mount /dev/vg_gluster/brick2 /bricks/brick2
```

Extend the */etc/fstab*:

```
/dev/vg_gluster/brick1 /bricks/brick1 xfs defaults 0 0
/dev/vg_gluster/brick2 /bricks/brick2 xfs defaults 0 0
```


Result: Both cluster nodes have two XFS partitions mounted to /bricks/brick{1,2} folders.

4.2. Trusted Pool (Storage Cluster)

Enable and start **glusterfsd.service** on both nodes:

```
# systemctl enable glusterd.service
ln -s '/usr/lib/systemd/system/glusterd.service' '/etc/systemd/system
/multi-user.target.wants/glusterd.service'

# systemctl start glusterd.service
```

 Ports TCP:24007-24008 are required for communication between **GlusterFS** nodes and each brick requires another TCP port starting at 24009.

Enable required ports on the firewall:

```
# firewall-cmd --zone=public --add-port=24007-24008/tcp --permanent
success


# firewall-cmd --reload
success
```

Use gluster command to connect the second **GlusterFS** node and create a Trusted Pool (Storage Cluster).

```
# gluster peer probe gluster2.example.com
peer probe: success.
```

Verify cluster peer:

```
# gluster peer status
Number of Peers: 1
Hostname: gluster2.example.com
Uuid: e528dc23-689c-4306-89cd-1d21a2153057
State: Peer in Cluster (Connected)
```

 Don't need to run "gluster peer probe" command on the second cluster node.

5. High Availability GlusterFS Volumes

GlusterFS Volume works with Gluster File System which is a logical collection of XFS bricks.

The following table shows dependencies between Volume types and sizes:

Available GlusterFS Volume types	Volume space calculations
Distributed (for maximum space)	$1G + 1G = 2G$
Replicated (for high availability)	$1G + 1G = 1G$
Striped (for large files)	$1G + 1G = 2G$
Distributed and Replicated	$(1G+1G) + (1G+1G) = 2G$
Distributed and Striped	$(1G+1G) + (1G+1G) = 4G$
Distributed, Replicated and Stripped	$[(1G+1G)+(1G+1G)] + [(1G+1G)+(1G+1G)] = 4G$

Open the required port on the firewall.

 Remember, each brick in the **GlusterFS** Volume requires a TCP port starting at 24009:

```
# firewall-cmd --zone=public --add-port=24009/tcp --permanent
```


```
success
```

```
# firewall-cmd --reload
success
```

Use the `/bricks/brick1` XFS partition on both nodes to create a **highly available Replicated Volume**. First create a sub-directory in `/bricks/brick1` mount point. It will be necessary for **GlusterFS**.

```
# mkdir /bricks/brick1/brick
```

Create a replicated **GlusterFS** Volume:

 Run this command on the first node *gluster1.example.com* only.

```
# gluster volume create glustervol1 replica 2 transport tcp
gluster1.example.com:/bricks/brick1/brick \
gluster2.example.com:/bricks/brick1/brick
volume create: glustervol1: success: please start the volume to
access data
```

```
# gluster volume start glustervol1
volume start: glustervol1: success
```

Verify **GlusterFS** Volumes:

```
# gluster volume info all
Volume Name: glustervol1
Type: Replicate
Volume ID: 6953a675-f966-4ae5-b458-e210ba8ae463
Status: Started
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: gluster1.example.com:/bricks/brick1/brick
Brick2: gluster2.example.com:/bricks/brick1/brick
Options Reconfigured:
performance.readdir-ahead: on
```

6. GlusterFS Clients

GlusterFS volumes can be accessed using **GlusterFS** Native Client (CentOS / RedHat / OracleLinux 6.5 or later), NFS v3 (other Linux clients), or CIFS (Windows clients).

6.1. Open the Firewall for Glusterfs/NFS/CIFS Clients

```
# firewall-cmd --zone=public --add-service=nfs --add-service=samba
--add-service=samba-client --permanent
success
```

```
# firewall-cmd --zone=public --add-port=111/tcp --add-port=139/tcp
--add-port=445/tcp --add-port=965/tcp --add-port=2049/tcp \
--add-port=38465-38469/tcp --add-port=631/tcp --add-port=111/udp
--add-port=963/udp --add-port=49152-49251/tcp --permanent
success
```

```
# firewall-cmd --reload
success
```

6.2. Access from another CentOS 7 machine via GlusterFS Native Client

All required packages are available by default in the CentOS 7 Base repository.

Install **GlusterFS** Client packages:

```
# yum install glusterfs glusterfs-fuse attr -y
```


Mount **GlusterFS** Volumes on the client:

```
# mount -t glusterfs gluster1.example.com:/glustervol1 /mnt/
```

Add a new line to the */etc/fstab* (optional):


```
gluster1.example.com:/glustervol1      /mnt  glusterfs
defaults,_netdev 0 0
```

6.3. Access from other Linux machines via NFS

 **GlusterFS** NFS server only supports version 3 of NFS protocol.

On both nodes, add the following line to the */etc/nfsmount.conf* file:

```
Defaultvers=3
```

 It is recommended to reboot all **glusterfs** nodes before continue.

Mount **GlusterFS** Volumes via NFS:

```
# mount -t nfs gluster1.example.com:/glustervol1 /mnt/
```

Append the following line to */etc/fstab* (optional):

```
gluster1.example.com:/glustervol1      /mnt  nfs  defaults,_netdev
0 0
```

6.4. Troubleshooting NFS

Issue:

```
# mount -t nfs gluster1.example.com:/glustervol1 /mnt/
mount.nfs: Connection timed out
```

The mount command fails because the NFS server is not running (N/A).

```
# gluster volume status
Status of volume: glustervol1
Gluster process                                TCP Port  RDMA
Port Online Pid
-----
Brick gluster1.example.com:/bricks/brick1/brick 49152
```

```

0          Y          2473
Brick gluster2.example.com:/bricks/brick1/brick  49152
0          Y          1394
NFS Server on localhost
N/A        N/A        N          N/A
Self-heal Daemon on localhost
N/A        N/A        Y          2562
NFS Server on gluster2.example.com
2049       0          Y          2531
Self-heal Daemon on gluster2.example.com
N/A        N/A        Y          2539

```

Solution:

Start/enable *rpcbind* service on the **GlusterFS** server.

```

# ps aux| grep rpcbind | grep -v grep
#

# /sbin/rpcbind -w

```

Restart **GlusterFS** Volume:

```

# gluster volume start glustervol1 force
volume start: glustervol1: success

```


6.5. Access from Windows/Linux machines via CIFS

Install/update the required packages on both cluster nodes:

```

# yum install samba samba-client samba-common samba-vfs-glusterfs
selinux-policy-targeted -y

```

 The samba package is installed previously

Start/enable Samba services.

```

# systemctl start smb.service
# systemctl enable smb.service
# systemctl start nmb.service
# systemctl enable nmb.service

```

Once a new **GlusterFS** Volume is created/started, it is added to the Samba configuration file automatically as *gluster-<Volume_name>* file share.

Verify **GlusterFS** shares in the */etc/samba/smb.conf*

```

[gluster-glustervol1]
comment = For samba share of volume glustervol1
vfs objects = glusterfs
glusterfs:volume = glustervol1
glusterfs:logfile = /var/log/samba/glusterfs-glustervol1.%M.log
glusterfs:loglevel = 7
path = /
read only = no
guest ok = yes

```

Add a new parameter - **kernel share modes = No** to the **GlusterFS** samba configuration.


```
[gluster-glustervol1]
comment = For samba share of volume glustervol1
vfs objects = glusterfs
glusterfs:volume = glustervol1
glusterfs:logfile = /var/log/samba/glusterfs-glustervol1.%M.log
glusterfs:loglevel = 7
path = /
read only = no
guest ok = yes
kernel share modes = No
```

Prepare the *glustervol1* **GlusterFS** Volume for Samba:

```
# gluster volume set glustervol1 stat-prefetch off
volume set: success

# gluster volume set glustervol1 server.allow-insecure on
volume set: success

# gluster volume set glustervol1 storage.batch-fsync-delay-usec 0
volume set: success
```

Add the following line to */etc/**glusterfs**/glusterd.vol* on each nodes:

```
option rpc-auth-allow-insecure on
```

Restart **glusterfs** service:

```
# systemctl restart glusterd.service
```

Define a new samba user:

```
# adduser sambauser

# smbpasswd -a sambauser
New SMB password:
Retype new SMB password:
Added user sambauser.
```

Set SELinux to enable sharing **GlusterFS** Volumes over Samba:

```
# setsebool -P samba_share_fusefs on
# setsebool -P samba_load_libgfapi on
```

Restart Samba:

```
# systemctl restart smb.service
# systemctl restart nmb.service
```

Mount **GlusterFS** Volume on Linux via CIFS (Samba):

```
# yum install cifs-utils -y

# mount -t cifs \\\gluster1.example.com\gluster-glustervol1 /mnt/
-o user=sambauser,pass=mypassword
```

Mount **GlusterFS** Volume on Windows:

```
c:\>net use Z: \\gluster1.example.com\gluster-glustervol1
/user:sambauser password
The command completed successfully.
```

6.6. Troubleshooting SELinux issues

Issue:

SELinux enforcing denies mount action to glusterd.

Log:

```
# tail -f /var/log/audit/audit.log

type=AVC msg=audit(1441281548.188:421): avc: denied { name_bind }
for pid=2636 comm="smbd" src=65534
scontext=system_u:system_r:smbd_t:s0
tcontext=system_u:object_r:unreserved_port_t:s0 tclass=tcp_socket
...
type=SYSCALL msg=audit(1441281548.188:421): arch=c000003e syscall=49
success=no exit=-13 a0=22 a1=7f3044bd1200 a2=10 a3=6 items=0
ppid=2201 pid=2636 auid=4294967295 uid=0 gid=0 euid=0 suid=0 fsuid=0
egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295 comm="smbd"
exe="/usr/sbin/smbd" subj=system_u:system_r:smbd_t:s0 key=(null)
...
type=AVC msg=audit(1441281548.188:422): avc: denied { name_connect
} for pid=2636 comm="smbd" dest=24007
scontext=system_u:system_r:smbd_t:s0
tcontext=system_u:object_r:gluster_port_t:s0 tclass=tcp_socket
...
type=SYSCALL msg=audit(1441281548.188:422): arch=c000003e syscall=42
success=no exit=-13 a0=22 a1=7f3044bd0d08 a2=10 a3=7f3041473c8c
items=0 ppid=2201 pid=2636 auid=4294967295 uid=0 gid=0 euid=0 suid=0
fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295
comm="smbd" exe="/usr/sbin/smbd" subj=system_u:system_r:smbd_t:s0
key=(null)
```

Solution:

```
# yum install policycoreutils-python -y

# setenforce 0

# load_policy
```

Now try to mount the Volume again.

 The policy will be enabled based on the error message.

Run the following commands to fix this SELinux issue:

```
# audit2allow -M glusterd_centos64 -l -i /var/log/audit/audit.log
***** IMPORTANT *****
To make this policy package active, execute:
semodule -i glusterd_centos64.pp

# setenforce 1
```

```
# semodule -i glusterd_centos64.pp
```


Reference: <http://blog.gluster.org/category/selinux/>

7. Extend GlusterFS Volumes without downtime

When expanding a distributed replicated or distributed striped volume, it needs to add a number of bricks that is a multiple of the replica or stripe count. For example, to expand a distributed replicated volume with a replica count of 2, it needs to add bricks in multiples of 2.

Volume parameters before the extension:

```
# gluster volume info all
Volume Name: glustervol1
Type: Replicate
Volume ID: 6953a675-f966-4ae5-b458-e210ba8ae463
Status: Started
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: gluster1.example.com:/bricks/brick1/brick
Brick2: gluster2.example.com:/bricks/brick1/brick
... output omitted ...
```

 Each brick requires another TCP port starting at 24009 so the port for the second brick will be the tcp/24010.

Open the firewall for the new brick:

```
# firewall-cmd --zone=public --add-port=24010/tcp --permanent
success

# firewall-cmd --reload
success
```

To extend the *glustervol1* Volume, use the remained two XFS bricks - created previously in the **XFS Bricks** section.

```
# mount |grep brick2
/dev/mapper/vg_gluster-brick2 on /bricks/brick2 type xfs
(rw,relatime,seclabel,attr2,inode64,noquota)
```

Create a necessary folder within */bricks/brick2*.

```
# mkdir /bricks/brick2/brick
```

Extend the **GlusterFS** Volume without any downtime:

```
# gluster volume add-brick glustervol1 gluster1.example.com:/bricks
/brick2/brick gluster2.example.com:/bricks/brick2/brick
volume add-brick: success
```


Verify the Volume:

```
# gluster volume info glustervol1
Volume Name: glustervol1
```

```
Type: Distributed-Replicate
Volume ID: 6953a675-f966-4ae5-b458-e210ba8ae463
Status: Started
Number of Bricks: 2 x 2 = 4
Transport-type: tcp
Bricks:
Brick1: gluster1.example.com:/bricks/brick1/brick
Brick2: gluster2.example.com:/bricks/brick1/brick
Brick3: gluster1.example.com:/bricks/brick2/brick
Brick4: gluster2.example.com:/bricks/brick2/brick
```

Now the Volume is extended with two bricks and became **Distributed-Replicate**.

Author: ZoltanPorkolab

HowTos/GlusterFSonCentOS (2015-09-04 16:40:30由  ZoltanPorkolab编辑)