

Firewall Ports

Port	Type	Description
24007	tcp	Glusterd communications
49152 - 59153	tcp	glusterfsd ports (one per brick)
111	tcp & udp	portmapper for NFS access
38465 - 38466	tcp	gluster nfs
11211	tcp & udp	memcached port for Swift
6000 - 6002	tcp	Swift Object, Container and Account server ports
443, 8080	tcp	Swift Proxy server

Upgrading to 3.4

Your version	Upgrade overview*
3.3	A rolling upgrade from 3.3 is possible, but a 'cold' upgrade is recommended. Once complete, an upgrade of all the native clients is recommended.
3.x	Downtime is required due to changes in the xlators and location of config files.

* Further details are available in gluster.org.

Recommended Configuration Limits

Min/Max Number of peers in a cluster	2/64
Clients per Volume	1000
Max Bricks per Node	4
Max bricksize (TB)	100

Scripting

Version 3.4 introduced the ability to generate command output in xml format using the --xml parameter.

The following example shows how this can be used from python to form a basis of server side automation scripts.

```
#!/usr/bin/env python
# Run the gluster command natively first to
# understand the xml layout.
#
# Invocation : <prog name> <volume name>
#

import xml.etree.ElementTree as ETree
import sys
from subprocess import Popen, PIPE

glfsCmd = Popen(['gluster',
                'vol',
                'status',
                sys.argv[1],
                'detail',
                '--xml'], stdout=PIPE)

# cmdOut will be a string object
cmdOut = glfsCmd.communicate()[0]

# Parse the string, making an xml object
xmlRoot = ETree.fromstring(cmdOut)

# Return a list of 'sizeTotal' elements
brickSize = xmlRoot.findall('./sizeTotal')

# Return a list of 'sizeFree' elements
brickFree = xmlRoot.findall('./sizeFree')

# Just count the number of 'path' elements in
# the XML to indicate the number of bricks in the
# volume
numBricks = len(xmlRoot.findall('./path'))

# Loop through each brickSize element, forming a
# new list of values, that are then sum'd
rawTotal = sum([float(thisBrick.text) for thisBrick in
brickSize])

rawFree = sum([float(thisBrick.text) for thisBrick in
brickFree])

pctUsed = ((rawTotal-rawFree)/rawTotal)

print "\nVolume Name: " + sys.argv[1]
print "Number of Bricks %5d" % (numBricks)
print "Raw Volume Size %5.02f (GB)" % (rawTotal/1024**3)
print "Raw Free %5.02f (GB)" %
(rawFree/1024**3)
print "% Used %5.02f\n" % (pctUsed)
```

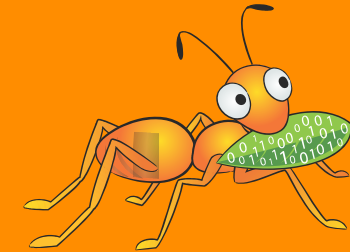
Useful links

Web

gluster.org forge.gluster.org

IRC Channel

irc.gnu.org#gluster



GLUSTER

Getting Started

with GlusterFS 3.4

Create pool → Define bricks → Create volume → Tune volume → Start volume

Creating and Managing Pools

Adding a node:

```
gluster peer probe NODE_NAME
```

Removing a node

```
gluster peer detach NODE_NAME
```

Querying status of the gluster (2 options)

```
gluster peer status
gluster pool list
```

Defining Bricks

Bricks should be configured with LVM (logical volume manager) for future flexibility and enhanced management. The following steps prepare an empty disk for use as a gluster brick, using /dev/sdb as an example device.

1. `pvcreate /dev/sdb`
2. `vgcreate VG_NAME /dev/sdb`
3. `lvcreate -n LV_NAME -l 100%PVS \ VG_NAME /dev/sdb`
4. `mkfs.xfs -i size=512 LV_PATH`

Once the logical volume is prepared, update fstab. Additional steps may be necessary if the disk device is a RAID LUN to ensure the device is aligned with the geometry of the underlying RAID group.

Creating and Managing Volumes

The process for creating a volume is as follows. First ensure the bricks are available.

1. `gluster vol create VOLNAME ...`
2. `gluster vol set VOLNAME KEY VALUE`
3. `gluster vol start VOLNAME`

Use “gluster vol help” for the complete syntax.

Expanding a volume

Distributed volumes may be expanded by any number of bricks, but replicated volumes must be expanded in units of

the replication factor (if a volume has a replication count 2, expansion must be in multiples of 2 bricks/nodes)

```
gluster vol add-brick VOLNAME BRICK(S)
```

Shrinking a volume

To remove bricks from a volume you *must use the start parameter* to avoid data loss!

```
gluster vol remove-brick VOLNAME BRICK start
```

Replacing a Brick

```
gluster vol add-brick VOLNAME NEW-BRICK
gluster vol remove-brick VOLNAME BRICK start
gluster vol remove-brick VOLNAME BRICK status
gluster vol remove-brick VOLNAME BRICK commit
```

Server Mount Options (fstab)

Filesystem	Options
xfs	alloccsize=4096, inode64, logbufs=8, logbsize=256K, noatime

Although ext4 is a very common Linux filesystem it is not recommended for production deployments of Gluster.

Client Mount Options (fstab)

Filesystem	Options	Required
gluster	_netdev	•
	backupvolfile-server=NODE enable-ino32	
cifs	_netdev, credentials=FILE	•
nfs	_netdev, vers=3, proto=tcp	•

Common Tuning Options

The following parameters are set via:

```
volume set VOL-NAME KEY VALUE
```

Key	Value and Action
nfs.disable	“on” turns NFS off
auth.allow or auth.reject	Supply IP addresses to permit or explicitly deny access to a volume

cluster.min-free-disk	% of free space to maintain across bricks
network.ping-timeout	Secs to wait before a node is declared 'dead'
user.cifs	Set to 'disable' to turn Samba off
storage.owner-uid X or storage.owner-gid X	Where 'x' is 360 → Virt/RHEV 161 → OpenStack Glance 165 → OpenStack Cinder
cluster.eager-lock	on (default) off Set to on to optimise lock — useful for high write workloads
nfs.enable-ino32	on off (default) Set to 'on' to enable 32bit nfs access to a gluster nfs volume

Use “gluster vol set help” for a more complete list of available options.

Using xattr's

Which bricks is my file stored on?

```
getfattr -d -e text -m . -n \
trusted.glusterfs.pathinfo \
FILE_PATH_NAME
```

Reusing a brick (after the volume is deleted)

```
setfattr -x trusted.gfid BRICK_PATH
setfattr -x trusted.glusterfs.volume-id \
BRICK_PATH
```

Cross Protocol Data Access

Although a gluster trusted pool can be configured to support multiple protocols simultaneously, a single volume can not be freely accessed by different protocols due to differences in locking semantics. The table below defines which protocols may safely access the same volume concurrently.

	SMB	NFS	Native	Object
SMB		✗	✗	✗
NFS	✗		✓	✓
Native	✗	✓		✓
Object	✗	✓	✓	