

GlusterFS это мощная сеть / кластер файловой системы исполняемая в пространстве пользователя, которая использует FUSE для подключения поверх реальной файловой системы. GlusterFS использует многоуровневый подход к файловой системе, где функции будут добавлены / удалены в соответствии с требованием. Хотя GlusterFS это виртуальная файловая система, она использует уже испытанные файловые системы на реальных дисках, такие как ext3 ext4, XFS, и т.д., чтобы хранить данные. Она может легко масштабироваться до петабайт дискового пространства, которое доступно пользователю под одной точкой монтирования.

Быстрый старт

Цель

Этот документ предназначен, чтобы дать вам шаг за шагом руководство по первой настройке GlusterFS. Для этого урока, мы предполагаем, что вы используете Fedora 20 в виртуальной машине. Мы также не объясняем здесь каждый шаг подробно, это руководство упрощено, чтобы помочь вам и дать возможность как можно быстрее начать работу с системой. После развертывания GlusterFS, выполните следующие действия, мы рекомендуем вам прочитать руководство GlusterFS администратора, чтобы узнать, как управлять GlusterFS и как выбрать тип тома, который соответствует вашим потребностям. Прочитайте руководство GlusterFS нового пользователя для более детального объяснения шагов, которые мы здесь предприняли. Мы хотим, чтобы вы успешно начали работать с системой в кратчайшие сроки.

Если вы хотели бы более подробную экскурсию по инструкциям установки GlusterFS, используя различные методы (в локальных виртуальных машинах, EC2 и baremetal) и различных дистрибутивах, то посмотрите ['Начать работать с GlusterFS'](#) и [Автоматическое развертывание GlusterFS с Puppet-Gluster + Vagrant](#)

Если вы хотите, развернуть GlusterFS автоматически, используя Puppet-Gluster + Vagrant, [посмотрите эту статью](#).

Реальное краткое руководство

Если вы хотите чистые МИНИМАЛЬНЫЕ действия по началу работы с GlusterFS, взгляните на это очень быстрое руководство.

Шаг 1 - Как минимум два узла

Fedora 20 на двух узлах под названием "server1" и "server2"

Рабочее подключение к сети

По два виртуальных диска, один для установки ОС, и один будет использоваться для обслуживания GlusterFS хранения SDB. Это будет эмулировать реальный мир развертывания, где вы хотели бы, чтобы GlusterFS - система хранения файлов отделялась от установки ОС.

Примечание: GlusterFS хранит свои динамически генерируемые файлы конфигурации в `/var/lib/glusterd`. Если в любой момент времени GlusterFS будет не в состоянии написать эти файлы, то это как минимум, вызовет странное поведение вашей системы; или еще хуже, получите системы в полностью автономном режиме. Желательно создать отдельные разделы для каталогов, таких как `/var/log/`, чтобы этого не произошло.

Шаг 2 — Форматируйте диски и установите точки монтирования

(На обоих узлах): Примечание: Эти примеры буду считать, что точка монтирования будет находиться на **/dev/sdb1**.

```
mkfs.xfs -i size=512 /dev/sdb1
mkdir -p /data/brick1
vi /etc/fstab
```

Добавьте следующее:

```
/dev/sdb1 /data/brick1 xfs defaults 1 2
```

Сохраните файл и выйдите, примонтируйте всё:

```
mount -a && mount
```

Вы должны увидеть **sdb1** установлен на **/data/brick1**

Шаг 3 - Установка GlusterFS

(на обоих серверах)

[Установка программного обеспечения](#)

```
yum install glusterfs-server
```

Стартуем демона GlusterFS :

```
service glusterd start
service glusterd status
glusterd.service - LSB: glusterfs server
  Loaded: loaded (/etc/rc.d/init.d/glusterd)
  Active: active (running) since Mon, 13 Aug 2012 13:02:11 -0700; 2s ago
  Process: 19254 ExecStart=/etc/rc.d/init.d/glusterd start (code=exited,
status=0/SUCCESS)
  CGroup: name=systemd:/system/glusterd.service
          └─ 19260 /usr/sbin/glusterd -p /run/glusterd.pid
             └─ 19304 /usr/sbin/glusterfsd --xlator-option georep-
server.listen-port=24009 -s localhost...
                └─ 19309 /usr/sbin/glusterfs -f /var/lib/glusterd/nfs/nfs-
server.vol -p /var/lib/glusterd/...
```

Шаг 4 - Настройка SELinux и IPTables

Это только в Fedora 19 или предыдущих версиях. Fedora 20 и позже должны работать нормально с SELinux

Изменить SELinux на "permissive" или "disabled" чтобы включить SELinux в разрешающий режим

```
setenforce 0
```

Чтобы увидеть текущий режим SELinux

```
getenforce
```

Для того, чтобы изменения SELinux сделать постоянными

В `"/etc/selinux/config"` вписать `"SELINUX=disabled"` или `"SELINUX=permissive"`

Удалить все **iptables** правила, так чтобы они не мешали Gluster

```
iptables -F
```

Шаг 5 - Настройка доверенного окружения

От "server1"

```
gluster peer probe server2
```

Примечание: При использовании хостов, первый сервер должен быть проверен с другого сервера, чтобы установить его имя.

От "server2"

```
gluster peer probe server1
```

Примечание: После того, как доверенное окружение было создано, только доверенные члены могут исследовать новые серверы в этом окружении. Новый сервер не может зондировать окружение, он должен быть проверен изнутри пула.

Шаг 6 - Настройка томов GlusterFS

На обоих server1 и server2:

```
mkdir /data/brick1/gv0
```

От любого сервера:

```
gluster volume create gv0 replica 2 server1:/data/brick1/gv0 server2:/data/brick1/gv0
```

```
gluster volume start gv0
```

Убедитесь, что тома показывают "Started":

```
gluster volume info
```

Примечание: Если работа нового тома не началась, то поможет понять, что пошло не так просмотр лог-файлов в `/VAR/log/GlusterFS` на одном или обоих серверах - как правило, в `etc-GlusterFS-glusterd.vol.log`

Шаг 7 - Проверка томов GlusterFS

На этом этапе мы будем использовать один из серверов для монтирования тома. Как правило, вы могли бы сделать это с внешнего аппарата, известного как "клиент". Для этого метода потребуются дополнительные пакеты, которые нужно установить на клиентском компьютере, серверы сейчас нам послужат для простого теста первого использования.

```
mount -t glusterfs server1:/gv0 /mnt
for i in `seq -w 1 100`; do cp -rp /var/log/messages /mnt/copy-test-$i; done
```

Во-первых, проверьте точку монтирования:

```
ls -lA /mnt | wc -l
```

Вы должны увидеть 100 файлов. Далее, проверьте в GlusterFS точки монтирования на каждом сервере:

```
ls -lA /data/brick1/gv0
```

Вы должны увидеть 100 файлов на серверах с помощью описанного способа. Без репликации, в пределах тома (здесь это не описано подробно), вы должны увидеть около 50 на каждом.

Оригинал:

QuickStart

Purpose

This document is intended to give you a step by step guide to setting up GlusterFS for the first time. For this tutorial, we will assume you are using Fedora 20 virtual machines (other distributions and methods can be found in the new user guide, below). We also do not explain the steps in detail here, this guide is just to help you get it up and running as soon as possible. After you deploy GlusterFS by following these steps, we recommend that you read the GlusterFS Admin Guide to learn how to administer GlusterFS and how to select a volume type that fits your needs. Read the GlusterFS New User Guide for a more detailed explanation of the steps we took here. We want you to be successful in as short a time as possible.

If you would like a more detailed walk through with instructions for installing using different methods (in local virtual machines, EC2 and baremetal) and different distributions, then have a look

at the [Getting Started](#) with GlusterFS guide.

Automatically Deploying GlusterFS with Puppet-Gluster+Vagrant

If you'd like to deploy GlusterFS automatically using Puppet-Gluster+Vagrant, have a look at [this article](#).

Really Really Quick Start Guide

If you want the bare minimum steps to getting started with GlusterFS, take a look at the [Really Really Quick Start Guide](#).

Step 1 – Have at least two nodes

- Fedora 20 on two nodes named "server1" and "server2"
- A working network connection
- At two virtual disks, one for the OS installation, and one to be used to serve GlusterFS storage (sdb). This will emulate a real world deployment, where you would want to separate GlusterFS storage from the OS install.
- Note: GlusterFS stores its dynamically generated configuration files at /var/lib/glusterd. If at any point in time GlusterFS is unable to write to these files, it will at minimum cause erratic behavior for your system; or worse, take your system offline completely. It is advisable to create separate partitions for directories such as /var/log to ensure this does not happen.

Step 2 - Format and mount the bricks

(on both nodes): Note: These examples are going to assume the brick is going to reside on /dev/sdb1.

```
mkfs.xfs -i size=512 /dev/sdb1
mkdir -p /data/brick1
vi /etc/fstab
```

Add the following:

```
/dev/sdb1 /data/brick1 xfs defaults 1 2
```

Save the file and exit

```
mount -a && mount
```

You should now see sdb1 mounted at /data/brick1

Step 3 - Installing GlusterFS

(on both servers) Install the software

```
yum install glusterfs-server
```

Start the GlusterFS management daemon:

```
service glusterd start
service glusterd status
glusterd.service - LSB: glusterfs server
```

```
Loaded: loaded (/etc/rc.d/init.d/glusterd)
Active: active (running) since Mon, 13 Aug 2012 13:02:11 -0700; 2s ago
Process: 19254 ExecStart=/etc/rc.d/init.d/glusterd start (code=exited,
status=0/SUCCESS)
CGroup: name=systemd:/system/glusterd.service
├─ 19260 /usr/sbin/glusterd -p /run/glusterd.pid
├─ 19304 /usr/sbin/glusterfsd --xlator-option georep-
server.listen-port=24009 -s localhost...
└─ 19309 /usr/sbin/glusterfs -f /var/lib/glusterd/nfs/nfs-
server.vol -p /var/lib/glusterd/...
```

Step 4 - Configure SELinux and iptables

This is only applicable to Fedora 19 or previous versions. Fedora 20 and later should work fine with SELinux

Change SELinux to either “permissive” or “disabled” mode To put SELinux in permissive mode

```
setenforce 0
```

To see the current mode of SELinux

```
getenforce
```

To make the SELinux changes permanent

Change “/etc/selinux/config” and make “SELINUX=disabled” or “SELINUX=permissive” in it

Remove all iptable rules, so that it does not interfere with Gluster

```
iptables -F
```

Step 5 - Configure the trusted pool

From "server1"

```
gluster peer probe server2
```

Note: When using hostnames, the first server needs to be probed from **one** other server to set its hostname.

From "server2"

```
gluster peer probe server1
```

Note: Once this pool has been established, only trusted members may probe new servers into the pool. A new server cannot probe the pool, it must be probed from the pool.

Step 6 - Set up a GlusterFS volume

On both server1 and server2:

```
mkdir /data/brick1/gv0
```

From any single server:

```
gluster volume create gv0 replica 2 server1:/data/brick1/gv0
server2:/data/brick1/gv0
gluster volume start gv0
```

Confirm that the volume shows "Started":

```
gluster volume info
```

Note: If the volume is not started, clues as to what went wrong will be in log files under /var/log/glusterfs on one or both of the servers - usually in etc-glusterfs-glusterd.vol.log

Step 7 - Testing the GlusterFS volume

For this step, we will use one of the servers to mount the volume. Typically, you would do this from an external machine, known as a "client". Since using the method here would require additional packages be installed on the client machine, we will use the servers as a simple place to test first.

```
mount -t glusterfs server1:/gv0 /mnt
for i in `seq -w 1 100`; do cp -rp /var/log/messages /mnt/copy-test-$i; done
```

First, check the mount point:

```
ls -lA /mnt | wc -l
```

You should see 100 files returned. Next, check the GlusterFS mount points on each server:

```
ls -lA /data/brick1/gv0
```

You should see 100 per server using the method we listed here. Without replication, in a distribute only volume (not detailed here), you should see about 50 each.

Wrap Up

So, that's it, about as fast as it can be shown. Make sure to have a look at the Admin Guide and New User guide to help you get a deeper understanding, but this is a great start to have a place to test things out.

