

University of Science and Technology of Hanoi



Distributed System

## **PRACTICAL WORK 6: GLUSTERFS**

### **Group members:**

Nguyen Phuong Thao (BI9-212)

Doan Tuyet Mai (BI9-162)

Trinh Thao Phuong (BI9-191)

Phung Kim Son (BI9-202)

Pham Minh Long (BI9-146)

Hanoi, Mar 2021

## 1. System organization

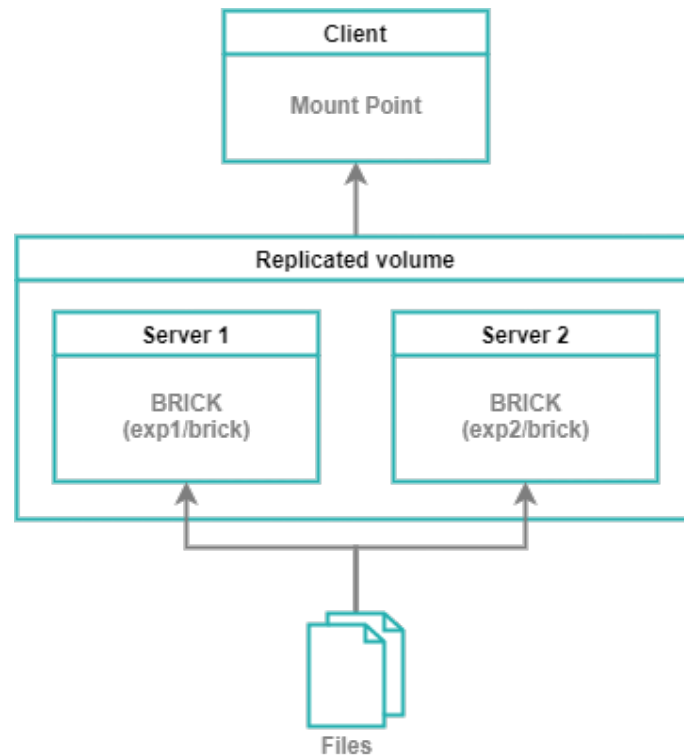


Figure 1: System organization

Gluster is a scalable, distributed file system that aggregates disk storage resources from multiple servers into a single global namespace. In this implementation, we created three virtual machine, two of them are servers which play the role of bricks in the replicated volume and a virtual machine is the client which is a mount point of the replicated volume. By default TCP protocol will be used by GlusterFS to transfer the files between the client and the volume.

## 2. Implementatation

In this labwork, we used the 3 virtual machines to act as the servers and client to perform the distributed replicated volume using GlusterFS. Each machine was set to 1 CPU, 1 GB of RAM and using Ubuntu 20.04 LTS, they connected each other via internal network.

### 2.1 Configuring the Host File

The first step we need to do before installing glusterfs on all servers is configuring the hosts' file.

---

```
sudo nano /etc/hosts
```

---

And then we add the IP of all servers, client to the host file of each machine. In this case, 2 servers (server,client) have the IP address 192.168.115.1 and 192.168.115.2, respectively. The client has the IP address 192.168.115.3

---

```
192.168.115.1 server
192.168.115.2 client
192.168.115.3 client-new
```

---

## 2.2 Installing GlusterFS Server, Configuring GlusterFS Servers

The second step is to install and configuring the GlusterFS in each server.  
To install GlusterFS, we follow the steps:

---

```
sudo apt install software-properties-common
sudo add-apt-repository ppa:gluster/glusterfs-7
sudo apt update
sudo apt install glusterfs-server
```

---

Next, in the both server, we need to enable the **glusterd** service as the following command

---

```
sudo systemctl start glusterd
sudo systemctl enable glusterd
```

---

From the server 1 (server), the server 2 (client) could be added to the storage trusted pool by the command

---

```
sudo gluster peer probe client
```

---

To check the storage pool status

---

```
sudo gluster pool list
```

---

And here is the output result

```
chaeyoung@server:~$ sudo gluster pool list
UUID                               Hostname      State
7fb59b51-3ac7-465e-ba89-cd7341d2d96e client        Connected
98848b3c-5e75-4408-b1e7-af1475445253 localhost     Connected
```

## 2.3 Setting up the Distributed GlusterFS Volume

First, we created new directory **/glusterfs/test** on both 2 servers

---

```
sudo mkdir -p /glusterfs/test
```

---

Then in the server 1 (server), we created the distributed glusterfs volume named **voltest** with 2 replicas **server** and **client**

---

```
sudo gluster volume create voltest transport tcp server:/glusterfs/test
client:/glusterfs/test force
```

---

Before accessing the data, we need to first start the volume

---

```
sudo gluster volume start voltest
```

---

And here is the information of the volume **voltest** after creation

```
chaeyoung@server:~$ sudo gluster volume info voltest
Volume Name: voltest
Type: Distribute
Volume ID: 8eb5d574-e68c-4ca8-a96d-785d9b267477
Status: Started
Snapshot Count: 0
Number of Bricks: 2
Transport-type: tcp
Bricks:
Brick1: server:/glusterfs/test
Brick2: client:/glusterfs/test
Options Reconfigured:
transport.address-family: inet
storage.fips-mode-rchecksum: on
nfs.disable: on
```

## 2.4 Configuring GlusterFS Client

First, we need to install `glusterfs-client` package.

Then, we created a new directory `/mnt/glusterfs` and then, mounted the distributed glusterfs volume `voltest` to the `/mnt/glusterfs` directory.

---

```
sudo mkdir -p /mnt/glusterfs
sudo mount -t glusterfs server:/voltest /mnt/glusterfs
```

---

And here is result

```
chaeyoung@client-new:~$ df -h /mnt/glusterfs
Filesystem      Size  Used Avail Use% Mounted on
server:/voltest 20G   7.5G   12G   40% /mnt/glusterfs
```

## 3. Benchmark

First, we created 2 test file, one is `file1.txt` with a small size of 6 bytes and `file3.txt` with a size of 524 KB.

---

```
sudo touch file1.txt file2.txt
```

---

And the following figures are the implementation of measuring the file I/O

```
chaeyoung@client-new:/mnt/glusterfs$ sudo dd if=file3.txt of=/mnt/file3.txt
1047+1 records in
1047+1 records out
536475 bytes (536 kB, 524 KiB) copied, 0.0347264 s, 15.4 MB/s

chaeyoung@client-new:/mnt/glusterfs$ sudo dd if=file1.txt of=/mnt/file1.txt
0+1 records in
0+1 records out
6 bytes copied, 0.002479 s, 2.4 kB/s _
```

As we can see here, the first text file has a very small size, as a result, the read speed is slow.

## 4. Contribution

| Student            | Student ID | Contribution             |
|--------------------|------------|--------------------------|
| Pham Minh Long     | BI9-146    | Write report             |
| Phung Kim Son      | BI9-202    | GlusterFS implementation |
| Trinh Thao Phuong  | BI9-191    | Write report             |
| Doan Tuyet Mai     | BI9-162    | Implement benchmark      |
| Nguyen Phuong Thao | BI9-212    | Research for GlusterFS   |