

Configuração GlusterFS Docker2

ChatGPT

Perfeito! Vamos criar uma configuração de **GlusterFS replicada entre os três servidores:**

- server04 → 147.93.176.48
- server02 → 212.56.34.241
- server03 → 212.56.34.172

Com os diretórios locais:

swift

Copiar Editar

```
/opt/swarm/npm/volumes/data  
/opt/swarm/npm/volumes/letsencrypt  
/opt/swarm/npm/volumes/mysql
```

Você quer que **cada um desses diretórios seja um volume replicado entre os três nós** — ou seja, 3 volumes distintos replicados 3x. Vamos fazer isso em etapas bem claras.

? 1. Configure `/etc/hosts` (em **todos os servidores**)

Para facilitar o uso de nomes ao invés de IPs diretos:

bash

Copiar Editar

```
echo "147.93.176.48 server04" | sudo tee -a /etc/hosts  
echo "212.56.34.241 server02" | sudo tee -a /etc/hosts  
echo "212.56.34.172 server03" | sudo tee -a /etc/hosts
```

? 2. Instale o GlusterFS (em todos os servidores)

Ubuntu/Debian:

bash

Copiar Editar

```
sudo apt update
sudo apt install glusterfs-server -y
sudo systemctl enable --now glusterd
```

CentOS/RHEL:

bash

Copiar Editar

```
sudo yum install centos-release-gluster
sudo yum install glusterfs-server -y
sudo systemctl enable --now glusterd
```

? 3. Verifique a conectividade entre os nós

Em todos os servidores, teste:

bash

Copiar Editar

```
ping server02
ping server03
ping server04
```

? 4. No `server04`: pareie os outros nós

No **server04**:

bash

Copiar Editar

```
gluster peer probe server02
gluster peer probe server03
```

Verifique:

bash

Copiar Editar

```
gluster peer status
```

Todos devem estar com status: Peer in Cluster (Connected)

? 5. Crie os diretórios de dados (em todos os 3 nós)

bash

Copiar Editar

```
sudo mkdir -p /opt/swarm/npm/volumes/data
sudo mkdir -p /opt/swarm/npm/volumes/letsencrypt
sudo mkdir -p /opt/swarm/npm/volumes/mysql
```

Recomendo garantir permissões para UID 1000 (usado por containers):

bash

Copiar Editar

```
sudo chown -R 1000:1000 /opt/swarm/npm/volumes
```

? 6. No `server04`: crie os volumes GlusterFS replicados

? Volume `npm-data`:

bash

Copiar Editar

```
gluster volume create npm-data replica 3 \
  server04:/opt/swarm/npm/volumes/data \
  server02:/opt/swarm/npm/volumes/data \
  server03:/opt/swarm/npm/volumes/data force
```

```
gluster volume start npm-data
```

? Volume `npm-letsencrypt`:

bash

Copiar Editar

```
gluster volume create npm-letsencrypt replica 3 \
  server04:/opt/swarm/npm/volumes/letsencrypt \
  server02:/opt/swarm/npm/volumes/letsencrypt \
```

```
server03:/opt/swarm/npm/volumes/letsencrypt force
```

```
gluster volume start npm-letsencrypt
```

? Volume npm-mysql:

```
bash
```

Copiar Editar

```
gluster volume create npm-mysql replica 3 \  
server04:/opt/swarm/npm/volumes/mysql \  
server02:/opt/swarm/npm/volumes/mysql \  
server03:/opt/swarm/npm/volumes/mysql force
```

```
gluster volume start npm-mysql
```

? 7. Verifique se os volumes foram criados corretamente

```
bash
```

Copiar Editar

```
gluster volume info
```

Você deve ver os 3 volumes com `Status: Started` e `Number of Bricks: 3 x 1 = 3`.

? 8. Monte os volumes no cliente Docker Swarm

No(s) nó(s) Swarm onde você quer usar os dados, monte os volumes:

```
bash
```

Copiar Editar

```
sudo mkdir -p /mnt/npm-swarm/data
```

```
sudo mkdir -p /mnt/npm-swarm/letsencrypt
```

```
sudo mkdir -p /mnt/npm-swarm/mysql
```

```
sudo mount -t glusterfs server04:/npm-data /mnt/npm-swarm/data
```

```
sudo mount -t glusterfs server04:/npm-letsencrypt /mnt/npm-swarm/letsencrypt
```

```
sudo mount -t glusterfs server04:/npm-mysql /mnt/npm-swarm/mysql
```

? 9. (Opcional) Automatize com /etc/fstab

Adicione ao `/etc/fstab` para montar automaticamente no boot:

fstab

Copiar Editar

```
server04:/npm-data /mnt/npm-swarm/data glusterfs defaults,_netdev 0 0
```

```
server04:/npm-letsencrypt /mnt/npm-swarm/letsencrypt glusterfs defaults,_netdev 0 0
```

```
server04:/npm-mysql /mnt/npm-swarm/mysql glusterfs defaults,_netdev 0 0
```

“ Teste com: `sudo mount -a`

? Conclusão

Você agora terá:

Volume GlusterFS	Diretório local montado	Usado por
<code>npm-data</code>	<code>/mnt/npm-swarm/data</code>	Nginx Proxy Mgr
<code>npm-letsencrypt</code>	<code>/mnt/npm-swarm/letsencrypt</code>	Nginx Proxy Mgr
<code>npm-mysql</code>	<code>/mnt/npm-swarm/mysql</code>	MariaDB

Se quiser, posso agora gerar seu `docker-stack.yml` já com esses volumes configurados para Swarm. Deseja?

Revision #1

Created 21 July 2025 22:06:21 by Administrador

Updated 21 July 2025 22:10:09 by Administrador