

Heartbeat - GlusterFS

*Vendredi 22/12/2023
Configuration de
Heartbeat et GlusterFS
Hakim BEN ATIA*

Configuration de la haute disponibilité

Ce guide détaille la mise en place d'une infrastructure de haute disponibilité pour WordPress sur Linux, en combinant Heartbeat, GlusterFS et l'utilisation de deux machines virtuelles distinctes. Les étapes détaillées incluent l'installation et la configuration de GlusterFS pour le stockage partagé, la mise en place du système LAMP, le déplacement des répertoires critiques vers cet espace partagé, l'installation spécifique de WordPress, et la configuration de Heartbeat pour assurer une disponibilité ininterrompue du service WordPress sur les deux machines. Ce processus vise à garantir la continuité et la fiabilité du site WordPress, même en cas de panne matérielle ou de défaillance d'une machine. Dans ce guide, nous vous guiderons à travers les étapes suivantes :

- 1- Changement des noms d'hôtes :** Modifiez les noms d'hôtes sur les deux machines virtuelles pour les identifier clairement dans le réseau.
- 2- Modification du fichier hosts :** Ajustez le fichier hosts sur les deux machines pour permettre la résolution DNS locale, facilitant ainsi la communication entre elles.
- 3- Installation de GlusterFS :** Installez GlusterFS sur les deux machines pour mettre en place le stockage partagé.
- 4- Configuration de GlusterFS et montage sur /mnt :** Configurez GlusterFS pour synchroniser les répertoires nécessaires et montez-les sur /mnt sur les deux machines pour une accessibilité uniforme.
- 5- Installation de LAMP :** Sur l'une des machines, installez et configurez l'environnement LAMP (Linux, Apache, MySQL, PHP) pour héberger le site WordPress.
- 6- Configuration de MariaDB et déplacement des données :** Configurez MariaDB pour diriger les données vers le répertoire monté /mnt de GlusterFS, assurant ainsi la disponibilité des données sur les deux machines.
- 7- Configuration du Virtual Host (vhost) pour Wordpress :** Configurez le vhost pour accueillir le site WordPress, en spécifiant le chemin monté via GlusterFS.
- 8- Configuration de la base de données de Wordpress :** Configurez la base de données MySQL/MariaDB pour le bon fonctionnement de WordPress.
- 9- Installation de Wordpress :** Installez WordPress sur l'environnement LAMP configuré, en spécifiant le chemin du site monté via GlusterFS pour garantir la haute disponibilité.
- 10- Installation et configuration de Heartbeat :** Installez Heartbeat pour surveiller la disponibilité des services et configurez-le pour maintenir la haute disponibilité entre les deux machines.

Prérequis

Pour réussir la configuration de la mise en haute disponibilité avec Heartbeat et GlusterFS sur Debian ou Ubuntu, assurez-vous de disposer des éléments suivants :

- 1- **Deux Machines Virtuelles (VM)** : Vous aurez besoin de deux VM distinctes.
- 2- **Système d'exploitation Debian ou Ubuntu** : Chaque VM doit être équipée d'une installation propre de Debian ou Ubuntu, de préférence la même version pour éviter les problèmes de compatibilité.
- 3- **Accès Administratif** : Vous devez disposer d'un accès administratif (par exemple, en utilisant un compte avec des privilèges sudo) sur chaque VM pour effectuer des configurations système.
- 4- **Connexion Réseau Fonctionnelle** : Assurez-vous que les VM sont correctement connectées au réseau, que les interfaces réseau sont configurées et que les VM peuvent communiquer entre elles.
- 5- **Accès Internet** : Si vous prévoyez de télécharger des packages ou des mises à jour, assurez-vous que vos VM ont accès à Internet.
- 6- **Connaissance avancée des systèmes** : Une compréhension approfondie des concepts systèmes est nécessaire pour suivre les étapes de configuration.

Conseil

Il est fortement recommandé de créer régulièrement des snapshots de vos machines virtuelles avant d'apporter des modifications majeures à leur configuration. Les snapshots permettent de sauvegarder l'état actuel de la VM, de sorte que vous puissiez facilement revenir en arrière en cas de problème ou d'erreur de configuration.

Utiliser des snapshots est une pratique courante pour éviter les erreurs coûteuses lors de la configuration et de l'expérimentation avec des machines virtuelles. Cela vous permet d'explorer et d'apprendre en toute sécurité sans risquer de perturber vos environnements de test.

Table des matières

Qu'est-ce que la Haute Disponibilité ?.....	5
Qu'est-ce que GlusterFS ?	5
Qu'est-ce que HeartBeat ?.....	5
Pourquoi mettre en place de la Haute Disponibilité ?.....	5
Commencer ?	5
Schéma réseau	6
Création des Deux VMs :.....	6
Configuration des Deux VMs	6
Installation de GlusterFS	8
Configuration de GlusterFS	8
Installation et configuration de la stack LAMP	11
Configuration Base de Données pour Wordpress	14
Installer et configurer Wordpress	14
Installation et configuration de HeartBeat.....	15
Votre serveur est Opérationnel !.....	17
Sources.....	17

Qu'est-ce que la Haute Disponibilité ?

La Haute Disponibilité (HD) dans les systèmes informatiques fait référence à la capacité d'un système ou d'une infrastructure à maintenir un niveau de fonctionnalité et d'accessibilité élevé, souvent proche de la continuité de service, même en cas de défaillance d'un ou plusieurs composants. Elle est généralement obtenue par la mise en place de dispositifs de redondance, de mécanismes de reprise sur incident, et de stratégies de répartition de charge pour minimiser les temps d'arrêt et assurer une accessibilité constante aux services critiques. Les composants clés incluent la duplication de matériel, l'utilisation de clusters, de solutions de stockage partagé, de mécanismes de basculement automatique et des politiques de surveillance active pour détecter et répondre rapidement aux pannes potentielles.

Qu'est-ce que GlusterFS ?

GlusterFS est un système de fichiers distribué qui fonctionne sur un réseau, permettant de stocker et de gérer d'importantes quantités de données sur plusieurs machines. Il offre une extensibilité en permettant d'ajouter facilement des nœuds pour augmenter la capacité de stockage, tout en assurant la disponibilité des données même en cas de panne matérielle. Son architecture peer-to-peer permet à chaque nœud du cluster d'accéder aux données, améliorant ainsi les performances et la résilience du système.

Qu'est-ce que HeartBeat ?

Heartbeat est un logiciel utilisé dans les environnements de haute disponibilité pour surveiller l'état des serveurs et des services. Il facilite la détection des pannes et la bascule automatique vers des systèmes redondants en cas de défaillance. Ce logiciel permet d'assurer la continuité des services critiques en orchestrant le passage transparent d'un serveur à un autre en cas de problème. Heartbeat est souvent utilisé conjointement avec d'autres outils pour garantir la disponibilité constante des applications et des services.

Pourquoi mettre en place de la Haute Disponibilité ?

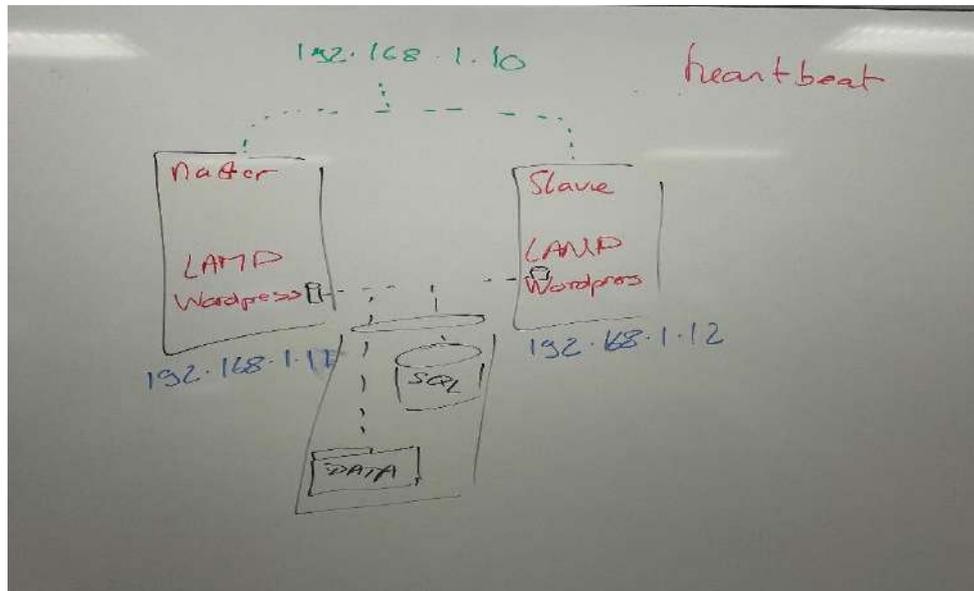
La mise en place de la Haute Disponibilité est cruciale pour garantir la continuité des services. Lorsque les systèmes sont répartis sur plusieurs machines ou serveurs, il est essentiel d'éviter les interruptions de service. La Haute Disponibilité assure cette continuité en permettant à un service de basculer vers un autre système disponible en cas de défaillance. Cela garantit que les services restent accessibles et opérationnels, offrant ainsi une expérience sans interruption pour les utilisateurs.

Commencer ?

Maintenant que nous avons clarifié l'objectif de cette documentation, nous pouvons commencer à explorer les étapes pour mettre en place de la Haute Disponibilité avec succès sur votre machine Linux.

Schéma réseau

Voici le schéma que nous utiliserons comme base pour ce tutoriel. Seules les adresses IP seront modifiées.



Création des Deux VMs :

Nous commençons par la création de deux machines virtuelles Debian, vous avez le choix entre Debian 11 et Debian 12 ou bien Ubuntu. Nous attribuerons un nom à chacune d'entre elles :

- 1- **Gluster1**
- 2- **Gluster2**

Nous les **laisserons configurées en mode NAT**, ce qui leur permettra d'accéder à Internet. Cette connectivité Internet est essentielle pour installer les différents services, effectuer les mises à jour et obtenir les outils nécessaires pour la configuration.

Configuration des Deux VMs

Ici je vais utiliser l'éditeur de texte par défaut « nano ».

On commence par une mise à jour des paquets du système :

- `apt-get update && apt-get upgrade -y`

Nous allons désormais passer en mode administrateur pour faciliter les commandes tout au long de ce tutoriel. Il est essentiel de noter que cette pratique est uniquement spécifique à ce guide. Il est déconseillé de s'y habituer, car une manipulation incorrecte peut compromettre le fonctionnement de votre machine.

- `su - root`

Nous allons attribuer un nom à chaque machine afin de simplifier l'exécution des commandes tout au long du tutoriel :

- Pour la machine 1 :
 - `hostnamectl set-hostname gluster1`
- Pour la machine 2 :
 - `hostnamectl set-hostname gluster2`

Rendez-vous ensuite dans le fichier suivant :

- `nano /etc/hosts`

À la fin de ce fichier, veuillez insérer les lignes suivantes :

- `192.168.135.135 gluster1`
- `192.168.135.129 gluster2`

Pour les explications :

- La première ligne indique que l'ip `192.168.135.135` est joignable au nom `gluster1`
- La deuxième ligne indique que l'ip `192.168.135.129` est joignable au nom `gluster2`

Une fois le fichier sauvegardé, vous pouvez le quitter.

Pour vérifier le bon fonctionnement, exécutez les commandes suivantes :

- A partir de la machine 2
 - `Ping gluster1`
- A partir de la machine 1
 - `Ping gluster2`

Voici les résultats attendus :

```
root@gluster1:~# ping gluster2
PING gluster2 (192.168.135.129) 56(84) bytes of data.
64 bytes from gluster2 (192.168.135.129): icmp_seq=1 ttl=64 time=0.304 ms
64 bytes from gluster2 (192.168.135.129): icmp_seq=2 ttl=64 time=0.345 ms
```

```
root@gluster2:~# ping gluster2
PING gluster2 (192.168.135.129) 56(84) bytes of data.
64 bytes from gluster2 (192.168.135.129): icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from gluster2 (192.168.135.129): icmp_seq=2 ttl=64 time=0.029 ms
64 bytes from gluster2 (192.168.135.129): icmp_seq=3 ttl=64 time=0.030 ms
```

La configuration initiale des deux machines est achevée. Nous allons désormais entamer l'installation et la configuration de GlusterFS sur ces deux machines.

Installation de GlusterFS

Ici je vais utiliser l'éditeur de texte par défaut « nano ».

On commence par installer l'outil :

- `apt-get install glusterfs-server -y`

À présent, nous allons initialiser l'outil et le configurer pour un démarrage automatique à chaque redémarrage de la machine :

- `systemctl start glusterd`
- `systemctl status glusterd` # Vérifie que le service est bien activé et en cours d'exécution
- `systemctl enable glusterd` # permet de lancer le service au boot de la vm (très important)

Configuration de GlusterFS

Pour commencer, créons le dossier `/data` qui agira comme point de référence pour les étapes suivantes :

- `cd && mkdir /data`

Ensuite **UNIQUEMENT** sur le `gluster1` (machine 1) effectuer la commande suivante :

- `gluster peer probe gluster2`

```
root@gluster1:~# gluster peer probe gluster2
peer probe: success
```

- `gluster peer status` # Afin de vérifier l'état du pool

```
root@gluster1:~# gluster peer status
Number of Peers: 1
Hostname: gluster2
Uuid: 5facef20-586c-455e-a232-815f7fc31cab
State: Peer in Cluster (Connected)
```

Nous allons procéder à la création du partage sur le `gluster1` (machine1) :

- `gluster volume create volume1 replica 2 transport tcp gluster1:/data/volume1 gluster2:/data/volume1 force`

```
root@gluster1:~# gluster volume create volume1 replica 2 transport tcp gluster1:/data/volume1 gluster2:/data/volume1 force
volume create: volume1: success: please start the volume to access data
```

Si tout est correct, passez à l'étape suivante : démarrez le volume pour le rendre opérationnel (UNIQUEMENT sur la machine 1) :

- gluster volume start volume1
- gluster volume info # Pour vérifier l'état du volume sur les deux machines

```
root@gluster2:~# gluster volume info
Volume Name: volume1
Type: Replicate
Volume ID: c1a488a5-3697-4118-9beb-d768d018ae42
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: gluster1:/data/volume1
Brick2: gluster2:/data/volume1
Options Reconfigured:
cluster.granular-entry-heal: on
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
```

```
root@gluster1:~# gluster volume info
Volume Name: volume1
Type: Replicate
Volume ID: c1a488a5-3697-4118-9beb-d768d018ae42
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: gluster1:/data/volume1
Brick2: gluster2:/data/volume1
Options Reconfigured:
cluster.granular-entry-heal: on
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
```

Sur les deux serveurs, procéder à l'installation du point de réplication sur /mnt en faisant :

- cd /mnt
- mkdir glusterfs

Le dossier de réplication étant créer nous allons donc monter la réplication sur les deux machines :

- Sur gluster1
 - mount.glusterfs gluster2:/volume1 /mnt/glusterfs
- Sur gluster2
 - mount.glusterfs gluster1:/volume1 /mnt/glusterfs

Vérifier si les directives ont bien été prises en compte sur les deux machines en faisant :

- df -h

```
root@gluster1:/mnt# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev              965M          0  965M   0% /dev
tmpfs             197M        692K  197M   1% /run
/dev/sda1         19G         2,7G   15G  16% /
tmpfs             984M          0  984M   0% /dev/shm
tmpfs             5,0M          0   5,0M   0% /run/lock
tmpfs             197M          0  197M   0% /run/user/1000
gluster2:/volume1 19G         2,9G   15G  16% /mnt/glusterfs
```

```
root@gluster2:/mnt# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev              965M          0  965M   0% /dev
tmpfs             197M        696K  197M   1% /run
/dev/sda1         19G         2,7G   15G  16% /
tmpfs             984M          0  984M   0% /dev/shm
tmpfs             5,0M          0   5,0M   0% /run/lock
tmpfs             197M          0  197M   0% /run/user/1000
gluster1:/volume1 19G         2,9G   15G  16% /mnt/glusterfs
```

Bravo ! Vous avez mis en place un stockage hautement disponible (HA) sur deux machines. Pour que cela fonctionne en permanence sans avoir à répéter la commande à chaque redémarrage, vous devez effectuer cette étape sur les deux machines :

- Sur le gluster1
 - nano /etc/fstab
 - gluster2:/volume1 /mnt/glusterfs glusterfs defaults,_netdev 0 0 # Insérer ceci à la fin

```

/etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=05e53917-c4ad-4286-ba47-1bdf63f1f373 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=3d0f6cbc-975b-4ac6-b677-7b9226238a4e none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0

gluster2:/volume1 /mnt/glusterfs glusterfs defaults,_netdev 0 0

```

- Sur le gluster2
 - nano /etc/fstab
 - gluster1:/volume1 /mnt/glusterfs glusterfs defaults,_netdev 0 0 # Insérer ceci à la fin

```

GNU nano 7.2 /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=05e53917-c4ad-4286-ba47-1bdf63f1f373 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=3d0f6cbc-975b-4ac6-b677-7b9226238a4e none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0

gluster1:/volume1 /mnt/glusterfs glusterfs defaults,_netdev 0 0

```

Pour vous assurer que les modifications sont prises en compte, redémarrez les deux machines, puis exécutez la commande suivante sur chacune d'entre elles :

- df -h

Cela devrait vous afficher ceci

```

root@gluster1:/mnt# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev                965M          0  965M   0% /dev
tmpfs               197M        692K  197M   1% /run
/dev/sda1           19G         2,7G   15G  16% /
tmpfs               984M          0  984M   0% /dev/shm
tmpfs               5,0M          0   5,0M   0% /run/lock
tmpfs              197M          0  197M   0% /run/user/1000
gluster2:/volume1  19G         2,9G   15G  16% /mnt/glusterfs

```

```

root@gluster2:/mnt# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev                965M          0  965M   0% /dev
tmpfs               197M        696K  197M   1% /run
/dev/sda1           19G         2,7G   15G  16% /
tmpfs               984M          0  984M   0% /dev/shm
tmpfs               5,0M          0   5,0M   0% /run/lock
tmpfs              197M          0  197M   0% /run/user/1000
gluster1:/volume1  19G         2,9G   15G  16% /mnt/glusterfs

```

Si vous obtenez les mêmes résultats, nous pouvons passer à la dernière étape de la partie GlusterFS qui est de tester la réplication entre les deux machines en effectuant les actions suivantes :

- Sur le Gluster1
 - cd /mnt/glusterfs # Pour se placer dans le dossier du réplica
 - touch helloworld salut bonjour # Pour créer des fichiers texte vide dans le répertoire
 - ls # Pour lister ce qu'il y a dans le dossier actuel

```
root@gluster1:/mnt/glusterfs# touch helloworld salut bonjour
root@gluster1:/mnt/glusterfs# ls
bonjour helloworld salut
```

Aller sur le gluster2 et vérifier si cela est bien présent dans le dossier de réplication :

- `cd /mnt/glusterfs`
- `ls`

Vous devriez apercevoir ceci

```
root@gluster2:~# cd /mnt/glusterfs/
root@gluster2:/mnt/glusterfs# ls
bonjour helloworld salut
```

Répétez la même procédure sur la deuxième machine. Si tout est correct, vous devriez observer les mêmes fichiers sur chacune des machines.

La configuration de GlusterFS est maintenant terminée. Votre système de réplication est fonctionnel, contribuant ainsi à garantir la haute disponibilité. En cas de panne d'une machine et de création ou modification de fichiers dans le répertoire `/mnt/glusterfs`, ces données resteront accessibles depuis l'autre machine active. Cela assure la continuité d'accès aux fichiers même en cas de panne d'une machine.

Installation et configuration de la stack LAMP

Ici je vais utiliser l'éditeur de texte par défaut « nano ».

Installer la stack LAMP (Linux, Apache2, MariaDB/MySQL, PHP) est relativement simple. Cependant, l'installation doit être réalisée dans un emplacement spécifique, celui que nous avons monté précédemment avec GlusterFS.

Sur les deux machines effectuer la commande suivante :

- `apt install apache2 mariadb-server php php-mysql -y`

UNIQUEMENT sur gluster1 faire :

- `systemctl stop mariadb # Pour stopper le service mariadb`
- `systemctl status mariadb # Pour vérifier qu'il est bien stopper`

Déplacer le dossier vers le replica via le gluster1 UNIQUEMENT :

- `mv /var/lib/mysql /mnt/glusterfs/mysql`
- `chown -R mysql:mysql /mnt/glusterfs/mysql/ # Pour le changement de propriétaire`
- `ls -l mysql/ # Pour afficher si tout à été bien pris en compte`

```
root@gluster1:/mnt/glusterfs# chown -R mysql:mysql /mnt/glusterfs/mysql
root@gluster1:/mnt/glusterfs# ls -l mysql/
total 111022
-rw-rw---- 1 mysql mysql 417792 7 janv. 15:40 aria_log.00000001
-rw-rw---- 1 mysql mysql 52 7 janv. 15:40 aria_log_control
-rw-r--r-- 1 mysql mysql 0 7 janv. 15:37 debian-10.11.flag
-rw-rw---- 1 mysql mysql 910 7 janv. 15:40 ib_buffer_pool
-rw-rw---- 1 mysql mysql 12582912 7 janv. 15:40 ibdata1
-rw-rw---- 1 mysql mysql 100663296 7 janv. 15:40 ib_logfile0
-rw-rw---- 1 mysql mysql 0 7 janv. 15:37 multi-master.info
drwx----- 2 mysql mysql 4096 7 janv. 15:37 mysql
-rw-r--r-- 1 mysql mysql 15 7 janv. 15:37 mysql_upgrade_info
drwx----- 2 mysql mysql 4096 7 janv. 15:37 performance_schema
drwx----- 2 mysql mysql 12288 7 janv. 15:37 sys
```

Sur les deux machines effectuer ceci :

- nano /etc/mysql/mariadb.conf.d/50-server.cnf

Rechercher la ligne « datadir »

```
#
# * Basic Settings
#
#user                = mysql
pid-file             = /run/mysqld/mysqld.pid
basedir              = /usr
#datadir              = /var/lib/mysql
#tmpdir               = /tmp
```

Modifier la ligne « datadir » en y ajoutant le nouveau lien vers le dossier mysql :

- datadir = /mnt/glusterfs/mysql

```
#
# * Basic Settings
#
#user                = mysql
pid-file             = /run/mysqld/mysqld.pid
basedir              = /usr
datadir              = /mnt/glusterfs/mysql
#tmpdir               = /tmp
```

- nano /etc/systemd/system/multi-user.target.wants/mariadb.service

Rechercher la ligne « ProtectHome »

```
# Doesn't yet work properly with SELinux enabled
# NoNewPrivileges=true

# Prevent accessing /home, /root and /run/user
ProtectHome=true
```

Modifier la ligne « ProtectHome » :

- ProtectHome= false

Puis recharger le daemon en faisant :

- systemctl daemon-reload

Vous avez terminé le déplacement du dossier MySQL et configuré les paramètres pour assurer le bon fonctionnement de MySQL en redirigeant ses fichiers de configuration vers la nouvelle destination.

Sur gluster1 UNIQUEMENT, vous allez configurer un VirtualHost pour accéder à WordPress en suivant ces étapes :

- a2dissite 000-default.conf # Afin de désactiver la page par défaut de Apache2
- nano /etc/apache2/sites-available/wordpress.conf

Dans le fichier wordpress.conf vous devrez insérer ceci :

```
<VirtualHost *:80>
    ServerName 192.168.135.135

    DocumentRoot /mnt/glusterfs/wordpress/
</VirtualHost>
```

Vous pouvez enregistrer et fermer le fichier, puis exécuter les commandes suivantes :

- a2ensite wordpress.conf # Pour activer le site wordpress
- systemctl restart apache2
- nano /etc/apache2/apache2.conf

```
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

Dans le fichier apache2.conf, ajustez les paramètres dans la section Directory et ajoutez "Allow from all" pour que cela ressemble à ceci :

```
# access here, or in any related virtual
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /mnt/glusterfs/wordpress>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
    Allow from all
</Directory>
```

Voici ce que vous devez mettre :

```
<Directory /mnt/glusterfs/wordpress>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
    Allow from all
</Directory>
```

Pour appliquer les modifications, redémarrez Apache2 et MySQL :

- systemctl restart apache2
- systemctl start mariadb

La configuration de la stack LAMP est désormais terminée.

Configuration Base de Données pour Wordpress

Cette étape est à effectuer uniquement sur le gluster1 :

- `sudo mysql -u root -p`

Appuyez sur Entrée pour passer le mot de passe car vous n'en avez pas définis un pour le moment.

Vous devez saisir les commandes suivantes :

- `CREATE DATABASE wordpress_db;`
- `CREATE USER 'wp_user'@'localhost' IDENTIFIED BY 'password';`
- `GRANT ALL ON wordpress_db.* TO 'wp_user'@'localhost' IDENTIFIED BY 'password';`
- `FLUSH PRIVILEGES;`
- `Exit;`

La configuration de la base de données WordPress sur le gluster1 est terminée. Passons à l'étape suivante.

Installer et configurer Wordpress

Toutes ses étapes sont à effectuer sur le gluster1 UNIQUEMENT :

- `cd /tmp`
- `wget https://wordpress.org/latest.tar.gz`
- `tar -xvf latest.tar.gz`
- `cp -R wordpress /mnt/glusterfs/`

En étant dans le dossier /tmp faire :

- `rm -rf latest.tar.gz wordpress`
- `cd /mnt/glusterfs/wordpress`
- `chown -R www-data:www-data /mnt/glusterfs/wordpress/`

```
root@gluster1:/mnt/glusterfs/wordpress# chown -R www-data:www-data /mnt/glusterfs/wordpress/
root@gluster1:/mnt/glusterfs/wordpress# ^C
root@gluster1:/mnt/glusterfs/wordpress# ls -l
total 201
-rw-r--r-- 1 www-data www-data 405 7 janv. 16:40 index.php
-rw-r--r-- 1 www-data www-data 19915 7 janv. 16:40 license.txt
-rw-r--r-- 1 www-data www-data 7399 7 janv. 16:40 readme.html
-rw-r--r-- 1 www-data www-data 7211 7 janv. 16:41 wp-activate.php
drwxr-xr-x 9 www-data www-data 4096 7 janv. 16:41 wp-admin
-rw-r--r-- 1 www-data www-data 351 7 janv. 16:40 wp-blog-header.php
-rw-r--r-- 1 www-data www-data 2323 7 janv. 16:41 wp-comments-post.php
-rw-r--r-- 1 www-data www-data 3613 7 janv. 16:40 wp-config-sample.php
drwxr-xr-x 4 www-data www-data 4096 7 janv. 16:40 wp-content
-rw-r--r-- 1 www-data www-data 5638 7 janv. 16:40 wp-cron.php
drwxr-xr-x 27 www-data www-data 12288 7 janv. 16:41 wp-includes
-rw-r--r-- 1 www-data www-data 2502 7 janv. 16:40 wp-links-opml.php
-rw-r--r-- 1 www-data www-data 3927 7 janv. 16:40 wp-load.php
-rw-r--r-- 1 www-data www-data 50924 7 janv. 16:40 wp-login.php
-rw-r--r-- 1 www-data www-data 8525 7 janv. 16:40 wp-mail.php
-rw-r--r-- 1 www-data www-data 26409 7 janv. 16:40 wp-settings.php
-rw-r--r-- 1 www-data www-data 34385 7 janv. 16:40 wp-signup.php
-rw-r--r-- 1 www-data www-data 4885 7 janv. 16:41 wp-trackback.php
-rw-r--r-- 1 www-data www-data 3154 7 janv. 16:40 xmlrpc.php
```

- `chmod -R 755 /mnt/glusterfs/wordpress/`

```
root@gluster1:/mnt/glusterfs/wordpress# chmod -R 755 /mnt/glusterfs/wordpress/
root@gluster1:/mnt/glusterfs/wordpress# ^C
root@gluster1:/mnt/glusterfs/wordpress# ls -l
total 201
-rwxr-xr-x 1 www-data www-data 405 7 janv. 16:40 index.php
-rwxr-xr-x 1 www-data www-data 19915 7 janv. 16:40 license.txt
-rwxr-xr-x 1 www-data www-data 7399 7 janv. 16:40 readme.html
-rwxr-xr-x 1 www-data www-data 7211 7 janv. 16:41 wp-activate.php
drwxr-xr-x 9 www-data www-data 4096 7 janv. 16:41 wp-admin
-rwxr-xr-x 1 www-data www-data 351 7 janv. 16:40 wp-blog-header.php
-rwxr-xr-x 1 www-data www-data 2323 7 janv. 16:41 wp-comments-post.php
-rwxr-xr-x 1 www-data www-data 3013 7 janv. 16:40 wp-config-sample.php
drwxr-xr-x 4 www-data www-data 4096 7 janv. 16:40 wp-content
-rwxr-xr-x 1 www-data www-data 5638 7 janv. 16:40 wp-cron.php
drwxr-xr-x 27 www-data www-data 12288 7 janv. 16:41 wp-includes
-rwxr-xr-x 1 www-data www-data 2502 7 janv. 16:40 wp-links-opml.php
-rwxr-xr-x 1 www-data www-data 3927 7 janv. 16:40 wp-load.php
-rwxr-xr-x 1 www-data www-data 50924 7 janv. 16:40 wp-login.php
-rwxr-xr-x 1 www-data www-data 8525 7 janv. 16:40 wp-mail.php
-rwxr-xr-x 1 www-data www-data 26409 7 janv. 16:40 wp-settings.php
-rwxr-xr-x 1 www-data www-data 34385 7 janv. 16:40 wp-signup.php
-rwxr-xr-x 1 www-data www-data 4885 7 janv. 16:41 wp-trackback.php
-rwxr-xr-x 1 www-data www-data 3154 7 janv. 16:40 xmlrpc.php
```

Votre installation et la configuration de WordPress sont terminées. Pour vérifier si tout fonctionne correctement, accédez à votre navigateur et entrez l'adresse de votre serveur gluster1 :

<http://192.168.135.135/>.



Après avoir sélectionné votre langue, vous devrez fournir les identifiants MySQL que vous avez configurés pour votre base de données WordPress précédemment.



Parfait, vous pouvez maintenant passer à l'étape suivante.

Installation et configuration de HeartBeat

L'installation de HeartBeat se fera sur les deux serveurs. Vous devrez donc copier-coller les mêmes fichiers de configuration sur les deux serveurs. On va ici créer un cluster HeartBeat avec deux serveurs membres de ce dernier (gluster1 et gluster2). Nous aurons donc un serveur actif en **192.168.135.135** et un serveur passif **192.168.135.129** tout deux sous Linux sur lesquels sera installé le paquet Heartbeat et ses dépendances. Nous souhaitons que les clients communiquent avec le serveur via l'IP virtuelle du cluster **192.168.135.133** et non pas

vers **192.168.135.135** ou **192.168.135.129**. Ce sera au cluster de passer les requêtes à tel ou tel serveur suivant la disponibilité du serveur "**maitre**".

Les fichiers de configuration devront être les mêmes sur les deux serveurs membres du cluster et devront se situés dans **"/etc/ha.d"** (ou **"/etc/heartbeat"** qui est un lien symbolique vers **"/etc/ha.d"**). Ces fichiers devront être créés car ils ne le sont pas à l'installation d'Heartbeat.

- apt install heartbeat -y
- nano /etc/ha.d/ha.cf

Dans le fichier ha.cf vous devrez mettre ceci :

```
# Indication du fichier de log
logfile /var/log/ha-log
# Les logs heartbeat seront gérés par syslog, dans la catégorie daemon
logfacility localo
# On définit la périodicité de contrôle des nœuds entre eux (en seconde)
keepalive 2
# Au bout de combien de seconde un nœud sera considéré comme "mort"
deadtime 10
# Quelle carte réseau utiliser pour les broadcasts Heartbeat (ens32 dans mon cas)
bcast ens32
# On liste tous les membres de notre cluster heartbeat (par les noms de préférences)
node gluster1
node gluster2
# Rebascule-t-on automatiquement sur le primaire si celui-ci redevient vivant ? oui*
auto_failback yes
respawn hacluster /usr/lib/heartbeat/ipfail
apiauth ipfail gid=haclient uid=hacluster
```

Enregistrer le fichier et quitter.

On passe maintenant au fichier **"/etc/heartbeat/haresources"** qui va contenir les actions à mener au moment d'un basculement. Plus clairement, quand un serveur passera du statuts "**passif**" à "**actif**", il ira lire ce fichier pour voir ce qu'il doit faire.

- nano /etc/heartbeat/haresources

Mettre dans le fichier « haresources » :

- gluster1 IPaddr::192.168.135.133/24/ens32 apache2 mysql

Sauvegardez les modifications et fermez le fichier.

Nous passons maintenant au fichier **"/etc/heartbeat/authkeys"**, ce fichier va définir la clé qui permettra d'authentifier un minimum les serveurs membres du cluster entre eux :

- nano /etc/heartbeat/authkeys

Inscrivez ces informations dans le fichier "authkeys" :

```
auth 1
1 sha1 MaClefSecrete
```

Sauvegardez les modifications et fermez le fichier.

- `chmod 600 /etc/heartbeat/authkeys # Sécuriser le fichier en mettant des droits plus restreints`

Vous avez achevé cette étape. Assurez-vous de répéter ces étapes sur les deux serveurs sans modifier la configuration ci-dessus.

Maintenant, démarrez le cluster et vérifiez les journaux en cas de problème :

- `/etc/init.d/heartbeat start`

Pour le fichier logs faites « `cat /var/log/ha-log` ».

Pour tester cela, éteignez votre machine principale. Vous constaterez que le service WordPress reprendra sur le serveur secondaire dès qu'il détecte que le serveur principal est hors ligne.

Votre serveur est Opérationnel !

Nous voilà à la conclusion de ce tutoriel dédié à la configuration de la haute disponibilité pour WordPress et MySQL, utilisant HeartBeat et GlusterFS sur deux serveurs Debian 12. Nous espérons que ce guide vous a été utile pour comprendre et mettre en place cette architecture cruciale pour garantir la disponibilité et la redondance de votre système.

En suivant les étapes détaillées de ce tutoriel, vous avez acquis des compétences précieuses dans la mise en place de la haute disponibilité pour des applications telles que WordPress et MySQL. Vous avez appris à utiliser HeartBeat pour surveiller les serveurs, à configurer GlusterFS pour la réplication des données, assurant ainsi une redondance efficace.

Nous vous encourageons à explorer davantage ces technologies pour personnaliser et optimiser la disponibilité de vos systèmes selon vos besoins spécifiques. La documentation demeure une ressource essentielle pour vous guider dans la configuration et la gestion de vos architectures haute disponibilité.

Si des questions subsistent ou si vous avez besoin d'une assistance supplémentaire, n'hésitez pas à explorer les ressources en ligne ou à consulter les forums de la communauté Debian. Nous vous remercions d'avoir suivi ce tutoriel et nous vous souhaitons le meilleur pour vos projets futurs de haute disponibilité sous Debian 12 !

Sources

- 1- <https://www.it-connect.fr/clustering-et-haute-disponibilite-sous-linux-avec-heartbeat%EF%BB%BF/>
- 2- <https://le-guide-du-secops.fr/2020/08/01/mise-en-place-dune-solution-de-replication-glusterfs-debian-ubuntu/>
- 3- <https://sio.loirecraft.com/s/2TfnjipawBmE8AH?dir=undefined&path=%2FSyst%C3%A8me%20%26%20service%2FHeartbeat%20-%20High%20Avaibility%20Linux&openfile=1359>