

Trevenque. Proyecto de formación dual.



Adrián Alonso Piñar
C.F.G.S Administración de Sistemas en Red
Proyecto Trevenque

Indice

| | |
|--|----|
| 1 Configuración del Servidor para Laravel, MariaDB y conexión entre ambas..... | 4 |
| 2 Implementación de Balanceo de Carga para MariaDB usando Galera Cluster..... | 9 |
| 3 Configuración de Balanceo de Carga para Laravel usando NGINX..... | 12 |

1 Configuración del Servidor para Laravel, MariaDB y conexión entre ambas

1.1. Actualización del sistema

En el servidor que alojará Laravel, comienza por actualizar los paquetes del sistema:

```
sudo apt update -y  
sudo apt upgrade -y
```

1.2. Instalación de PHP y extensiones necesarias

Laravel necesita PHP y varias extensiones. Instálalos con:

```
sudo apt install php php-cli php-mbstring php-xml php-bcmath php-curl php-mysql php-zip unzip curl git -y
```

1.3. Instalación de Composer

Composer es el gestor de dependencias de PHP:

```
curl -sS https://getcomposer.org/installer | php  
sudo mv composer.phar /usr/local/bin/composer
```

1.4. Instalación del servidor web Apache

```
sudo apt install apache2 libapache2-mod-php -y
```

1.5. Creación del proyecto Laravel

```
cd /var/www
```

```
sudo composer create-project --prefer-dist laravel/laravel laravel
```

1.6. Asignación de permisos

```
sudo chown -R www-data:www-data laravel
sudo chmod -R 775 laravel/storage
sudo chmod -R 775 laravel/bootstrap/cache
```

1.7. Configuración del VirtualHost de Apache

Crea el archivo de configuración:

```
sudo nano /etc/apache2/sites-available/laravel.conf
```

```
<VirtualHost *:80>
    ServerName laravel.local
    DocumentRoot /var/www/laravel/public

    <Directory /var/www/laravel/public>
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/laravel_error.log
    CustomLog ${APACHE_LOG_DIR}/laravel_access.log combined
</VirtualHost>
```

Activa el sitio y el módulo mod_rewrite:

```
sudo a2ensite laravel.conf  
sudo a2enmod rewrite
```

Edita el archivo principal de configuración de Apache:

```
sudo nano /etc/apache2/apache2.conf
```

```
<Directory /var/www/>  
    AllowOverride All  
    Require all granted  
</Directory>
```

Reinicia Apache:

```
sudo systemctl restart apache2
```

Parte 2: Configuración del Servidor para MariaDB

2.1. Actualización del sistema

```
sudo apt update -y  
sudo apt upgrade -y
```

2.2. Instalación de MariaDB

```
sudo apt install mariadb-server -y
```

2.3. Configuración de seguridad

```
sudo mysql_secure_installation
```

2.4. Habilitar acceso remoto

Edita la configuración de red del servidor MariaDB:

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

Modificaremos:

```
bind-address = 127.0.0.1
```

Por:

```
bind-address = 0.0.0.0
```

Reinicia el servicio para aplicar los cambios:

```
sudo systemctl restart mariadb
```

2.5. Crear base de datos y usuario para Laravel

```
sudo mariadb
```

```
CREATE DATABASE laravel_db;
```

```
CREATE USER 'laravel_user'@'%' IDENTIFIED BY 'tu_clave_segura';
```

```
GRANT ALL PRIVILEGES ON laravel_db.* TO 'laravel_user'@'%';
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

Parte 3: Conexión de Laravel a MariaDB

3.1. Configurar archivo .env

Edita el archivo .env del proyecto Laravel:

```
DB_CONNECTION=mysql
```

```
DB_HOST=IP_DEL_SERVIDOR_MARIADB
```

```
DB_PORT=3306  
DB_DATABASE=laravel_db  
DB_USERNAME=laravel_user  
DB_PASSWORD=tu_clave_segura
```

3.2. Probar la conexión

Ejecuta el siguiente comando para verificar la conexión:

```
php artisan migrate
```

Si desde el buscador, entramos ahora a la IP de la maquina, podremos ver, si lo hemos realizado todo correctamente, que nos carga la pagina de laravel.

2 Implementación de Balanceo de Carga para MariaDB usando Galera Cluster

1. Actualización del sistema

Ejecutar en cada nodo:

```
sudo apt update -y  
sudo apt upgrade -y
```

2. Instalación de MariaDB y Galera

Instalar MariaDB junto con Galera en cada nodo:

```
sudo apt install mariadb-server galera-4 -y
```

3. Configuración de Galera

Editar el archivo de configuración:

```
sudo nano /etc/mysql/mariadb.conf.d/60-galera.cnf
```

Agregar o modificar las siguientes líneas (ajustar según cada nodo):

```
[galera]
wsrep_cluster_name = "galera-cluster"
wsrep_cluster_address = gcomm://10.211.20.150,10.211.20.151,10.211.20.152
wsrep_node_name = galera-node1 # Reemplazar según el nodo: galera-node2,
galera-node3
wsrep_node_address = 10.211.20.150 # Dirección IP local correspondiente
wsrep_on = ON
wsrep_provider = /usr/lib/galera/libgalera_smm.so
wsrep_sst_method = rsync
```

4. Apertura de puertos para la comunicación entre nodos

Ejecutar en todos los nodos:

```
sudo ufw allow 3306/tcp # MySQL
sudo ufw allow 4567/tcp # Galera replication
sudo ufw allow 4568/tcp # Incremental State Transfer
sudo ufw allow 4444/tcp # State Snapshot Transfer
```

5. Inicialización del clúster

Este paso se realiza solo en el primer nodo del clúster:

```
sudo systemctl stop mariadb
sudo galera_new_cluster
```


Verificar el estado del clúster:

```
mysql -u root -p -e "SHOW STATUS LIKE 'wsrep_cluster_size';"
```

Como resultado debe salir "1" si funciona correctamente.

6. Conexión de los nodos restantes

En los nodos 2 y 3:

```
sudo systemctl stop mariadb
```

```
sudo systemctl start mariadb
```

Verificar el tamaño del clúster desde cualquier nodo:

```
mysql -u root -p -e "SHOW STATUS LIKE 'wsrep_cluster_size';"
```

Como resultado debe salir "3" si funciona correctamente.

7. Prueba rápida de replicación

Desde cualquier nodo, crea una base de datos:

```
CREATE DATABASE prueba_galera;
```

Luego, en los otros nodos, ejecutar:

```
SHOW DATABASES;
```

La base de datos `prueba_galera` debe estar presente en todos los nodos, lo que confirma que la replicación funciona correctamente

3 Configuración de Balanceo de Carga para Laravel usando NGINX

Arquitectura

La estructura propuesta se basa en los siguientes componentes:

| Maquinas | IPs |
|---------------------|---------------|
| Laravel1 | 10.211.20.100 |
| Laravel2 | 10.211.20.101 |
| Nginx Load Balancer | 10.211.20.200 |

Entra al servidor de Nginx y haz lo siguiente:

1. Instalación de NGINX

Actualizar el sistema e instalar NGINX:

```
sudo apt update -y
sudo apt upgrade -y
sudo apt install nginx -y
```

2. Configuración del sitio para balanceo

Crear el archivo de configuración de NGINX:

```
sudo nano /etc/nginx/sites-available/laravel-lb
```

Que debe llevar el archivo:

```
upstream laravel_cluster {
    server 10.211.20.100;
    server 10.211.20.101;
}

server {
    listen 80;

    location / {
        proxy_pass http://laravel_cluster;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

3. Activación de la configuración

Eliminar el sitio por defecto y habilitar la nueva configuración:

```
sudo rm /etc/nginx/sites-enabled/default
sudo ln -s /etc/nginx/sites-available/laravel-lb /etc/nginx/sites-enabled/
```

Probar la configuración y recargar NGINX:

```
sudo nginx -t
sudo systemctl reload nginx
```

Finalmente, para probar que funciona, debemos entrar desde el buscador en la IP del servidor de Nginx en este caso la "10.211.20.200", si todo funciona correctamente deberá redirigirte a la pagina de Laravel.