

Replicación de bases de datos

TAGS

[DATABASES](#),

Una parte fundamental de este proyecto es la replicación de la base de datos que consiste en un sistema de base de datos distribuido el cual consistirá en una base maestra y una base esclava que estarán contenerizadas (dockerizadas) que usaran el modelo de replicación que la base esclava copiara en tiempo real a la base maestra.

Primero creamos el docker-compose con dos servicios los cuales serán la data-base maestra y la data-base esclava creadas con postgresql 11 con volúmenes permanentes

Configuración del docker

```
version: '3'

services:
  postgresql-master:
    image: 'docker.io/bitnami/postgresql:11-debian-10'
    container_name: postgres-master
    ports:
      - '5432:5432'
    volumes:
      - 'postgresql_master_data:/bitnami/postgresql'
    environment:
      - POSTGRESQL_PGAUDIT_LOG=READ,WRITE
      - POSTGRESQL_LOG_HOSTNAME=true
      - POSTGRESQL_REPLICATION_MODE=master
      - POSTGRESQL_REPLICATION_USER=repl_user
      - POSTGRESQL_REPLICATION_PASSWORD=repl_password
      - POSTGRESQL_USERNAME=postgres
      - POSTGRESQL_DATABASE=my_database
      - ALLOW_EMPTY_PASSWORD=yes

  postgresql-slave:
    image: 'docker.io/bitnami/postgresql:11-debian-10'
    container_name: postgres-slave
    ports:
      - '5433:5432'
    depends_on:
      - postgresql-master
    environment:
      - POSTGRESQL_USERNAME=postgres
      - POSTGRESQL_PASSWORD=my_password
      - POSTGRESQL_MASTER_HOST=postgresql-master
      - POSTGRESQL_PGAUDIT_LOG=READ,WRITE
```

```
- POSTGRESQL_LOG_HOSTNAME=true
- POSTGRESQL_REPLICATION_MODE=slave
- POSTGRESQL_REPLICATION_USER=repl_user
- POSTGRESQL_REPLICATION_PASSWORD=repl_password
- POSTGRESQL_MASTER_PORT_NUMBER=5432
```

```
volumes:
  postgresql_master_data:
  driver: local
```

Ahora levantamos el docker para que funcione

```
docker-compose up
```

Después accedemos al contener de la base maestra

```
docker-compose exec postgresql-master bash
psql -U postgres
```

Ahora creamos un usuario llamado usuario de lectura y garantizamos todos los permisos al usuario de lectura

```
CREATE USER reading_user WITH PASSWORD 'reading_pass';
GRANT CONNECT ON DATABASE my_database TO reading_user;
\connect my_database
GRANT SELECT ON ALL TABLES IN SCHEMA public TO reading_user;
GRANT SELECT ON ALL SEQUENCES IN SCHEMA public TO reading_user;
GRANT USAGE ON SCHEMA public TO reading_user;
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO reading_user;
```

Para poder confirmar que esta funcionando correctamente una vez adentro en la base de datos maestra creamos una tabla de pruebas y la llenamos de datos para comprobar que este funcionando correctamente

```
CREATE TABLE test_table (
  column1 text,
  column2 text
);
INSERT INTO test_table (column1, column2) VALUES ('data1', 'data2');
postgres=# SELECT * FROM test_table;
 column1 | column2
-----+-----
 data1   | data2
(1 row)
```

Ahora nos metemos a la base esclava y comprobamos que la tabla exista

```
docker-compose exec postgresql-slave bash  
psql -U postgres
```

```
postgres=# SELECT * FROM test_table;  
 column1 | column2  
-----+-----  
 data1   | data2  
(1 row)
```

y al final eliminamos la tabla en la base maestra en base maestra para limpiar la base maestra para limpiar la bases

```
DROP TABLE test_table;
```