

Práctica 5

El desarrollo de esta práctica tiene un objetivo muy claro y es la replicación de bases de datos en MySQL, de forma que tendremos un servidor principal, el maestro, en nuestro caso M1 y un servidor secundario de backup o esclavo, M2 (como opcional se propone la configuración maestro-maestro). Para lograr esto, en primer lugar debemos configurar iptables para que la comunicación entre las máquinas M1 y M2 sea posible, para ello añadimos al script de iptables (realizado durante la práctica anterior) la siguiente línea:

```
iptables -A INPUT -s 192.168.56.102 -j ACCEPT
```

De esta forma aceptaremos cualquier comunicación proveniente de la M2. Así mismo, añadiremos en el script de M2 la misma línea cambiando la IpM2 por IpM1 de forma que la M2 acepte todas las comunicaciones provenientes de M1.

Ahora comprobaremos que mysql está correctamente instalado ejecutando el comando:

```
- sudo mysql -u root -p
```

Tendremos que introducir la contraseña que le dimos en el momento de su instalación y crearemos la base de datos estudiante con la tabla de datos:

```
root@UbuntuServer:/home/laguilarg99# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.23-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database estudiante;
Query OK, 1 row affected (0,00 sec)

mysql> use estudiante
Database changed
mysql> show tables
+----+
|  |
+----+
Empty set (0,00 sec)

mysql> create table datos(nombre varchar(100), apellidos varchar(100), usuario varchar(100), email varchar(100));
Query OK, 0 rows affected (2,70 sec)

mysql> show tables;
+-----+
| Tables_in_estudiante |
+-----+
| datos                 |
+-----+
1 row in set (0,00 sec)
```

Ahora introduciremos nuestros datos como prueba de que la estructura creada los almacena correctamente:

```
mysql> insert into datos(nombre,apellidos,usuario,email) values ("Luis Miguel","Aguilar González"
"laguilarg99", "laguilarg99@correo.ugr.es");
Query OK, 1 row affected (0,33 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos | usuario | email |
+-----+-----+-----+-----+
| Luis Miguel | Aguilar González | laguilarg99 | laguilarg99@correo.ugr.es |
+-----+-----+-----+-----+
1 row in set (0,00 sec)
```

```
mysql> describe datos;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nombre | varchar(100) | YES | | NULL | |
| apellidos | varchar(100) | YES | | NULL | |
| usuario | varchar(100) | YES | | NULL | |
| email | varchar(100) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0,00 sec)
```

El siguiente paso sería replicar la BD a la segunda máquina, para eso usaremos mysqldump.

En la máquina M1 ejecutamos:

```
mysql> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0,00 sec)

mysql> QUIT
Bye
```

```
root@UbuntuServer:/home/laguilarg99# sudo mysqldump estudiante -u root -p > /tmp/estudiante.sql
Enter password:
root@UbuntuServer:/home/laguilarg99# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.23-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> UNLOCK TABLES;
Query OK, 0 rows affected (0,00 sec)
```

Haciendo uso de scp copiamos el archivo sql de una máquina a otra:

```
root@UbuntuServer:/home/laguilarg99# sudo scp /tmp/estudiante.sql laguilarg99@192.168.56.102:/tmp/es-
tudiante.sql
laguilarg99@192.168.56.102's password:
estudiante.sql                                100% 2007      2.0KB/s   00:00
```

Ahora en la máquina M2 ejecutamos lo siguiente:

```
root@UbuntuServer:/home/laguilarg99# sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.23-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database estudiante;
Query OK, 1 row affected (0,00 sec)

mysql> quit
Bye
```

Es en este momento cuando entra en juego el fichero .sql de la máquina M1, para ello ejecutamos

- `sudo mysql -u root -p estudiante < /tmp/estudiante.sql`

Para demostrar que ha funcionado en M2 desde la línea de comandos de mysql, ejecutamos:

```
mysql> use estudiante
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos | usuario | email |
+-----+-----+-----+-----+
| Luis Miguel | Aguilar González | laguilarg99 | laguilarg99@correo.ugr.es |
+-----+-----+-----+-----+
1 row in set (0,00 sec)
```

Para realizar una configuración Maestro-esclavo, en el archivo de configuración deberemos comentar la línea, *#bind-address*:

```
#bind-address = 127.0.0.1
```

Y descomentar *server-id = 1*, la línea que contiene *log_error* la dejamos igual, por otro lado el registro binario *log_bin*, lo descomentamos:

```
log_bin = /var/log/mysql/mysql-bin.log
```

Guardamos el fichero y reiniciamos el servicio:

- `systemctl restart mysql`

En M2 replicamos los pasos a excepción de que **server-id = 2** y de nuevo reiniciamos el servicio.

En M1, dentro de mysql realizamos las siguientes operaciones:

```
mysql> CREATE USER esclavo IDENTIFIED BY 'esclavo';
Query OK, 0 rows affected (0,16 sec)

mysql> GRANT REPLICATION SLAVE ON *.* TO 'esclavo'@'%' IDENTIFIED BY 'esclavo';
Query OK, 0 rows affected, 1 warning (0,15 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,11 sec)

mysql> FLUSH TABLES;
Query OK, 0 rows affected (0,15 sec)

mysql> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0,00 sec)

mysql> SHOW MASTER STATUS;
+-----+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
+-----+-----+-----+-----+-----+
| mysql-bin.000001 |      980 |              |                  |                  |
+-----+-----+-----+-----+-----+
1 row in set (0,00 sec)
```

En M2, dentro de mysql ejecutamos:

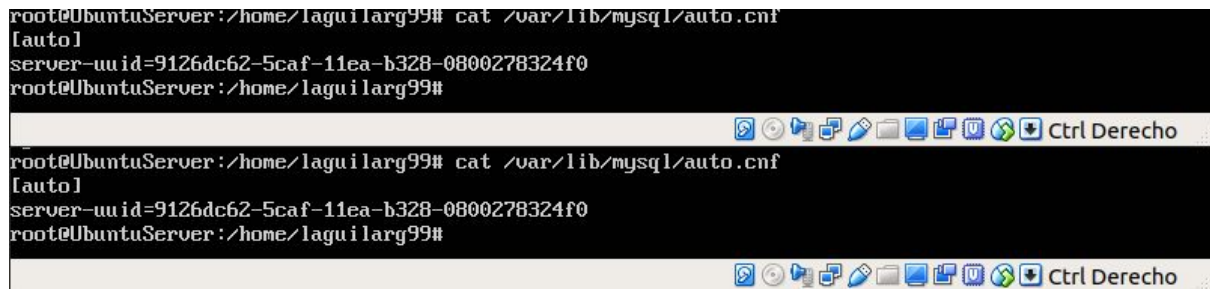
```
CHANGE MASTER TO MASTER_HOST='192.168.56.101', MASTER_USER='esclavo',
MASTER_PASSWORD='esclavo',MASTER_LOG_FILE='mysql-bin.000001',
MASTER_LOG_POS=980, MASTER_PORT=3306;
```

Tras ejecutar: `SHOW SLAVE STATUS\G`

```
Last_IO_Errno: 1593
Last_IO_Error: Fatal error: The slave I/O thread stops because master and slave have
equal MySQL server UIDs; these UIDs must be different for replication to work.
```

Este error aparece debido a que ambas máquinas son clonaciones y por tanto tienen el mismo UUID como se puede observar:

```
root@UbuntuServer:/home/laguilarg99# cat /var/lib/mysql/auto.cnf
[auto]
server-uuid=9126dc62-5caf-11ea-b328-0800278324f0
root@UbuntuServer:/home/laguilarg99#
```

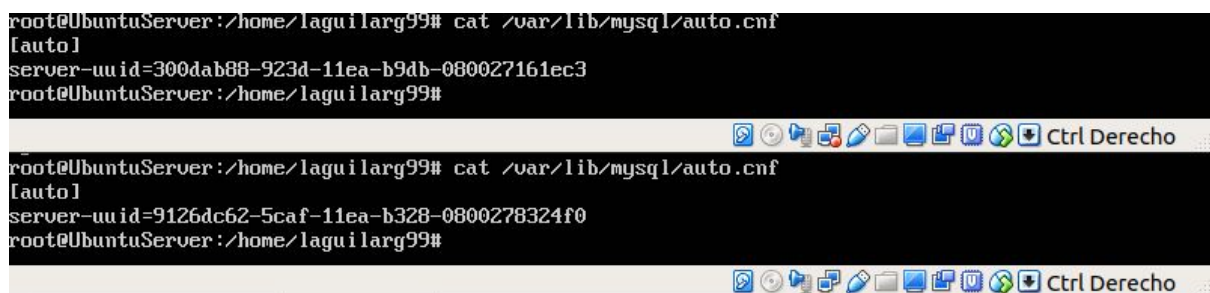


Para solucionarlo simplemente borraremos alguno de los dos y reiniciamos el servicio de mysql.

- `rm -rf /var/lib/mysql/auto.conf`
- `systemctl restart mysql`

Como se puede observar, ahora ambos UUID son diferentes:

```
root@UbuntuServer:/home/laguilarg99# cat /var/lib/mysql/auto.cnf
[auto]
server-uuid=300dab88-923d-11ea-b9db-080027161ec3
root@UbuntuServer:/home/laguilarg99#
```



Ahora al ejecutar `SHOW SLAVE STATUS\G` no aparece error alguno.

```
Seconds_Behind_Master: 0
Master_SSL_Verify_Server_Cert: No
Last_IO_Errno: 0
Last_IO_Error:
```

Para que la segunda máquina funcione de maestro al igual que la primera he replicado los pasos, en la máquina M2 creamos el usuario esclavo y mostramos el estado MASTER:

```
mysql> SHOW MASTER STATUS;
```

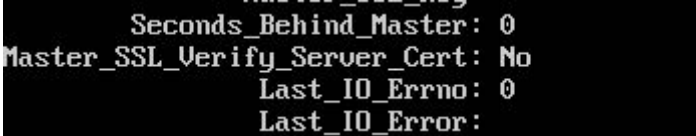
File	Position	Binlog_Do_DB	Binlog_Ignore_DB	Executed_Gtid_Set
mysql-bin.000002	980			

```
1 row in set (0.00 sec)
```

En la máquina M1:

```
CHANGE MASTER TO MASTER_HOST='192.168.56.102', MASTER_USER='esclavo',  
MASTER_PASSWORD='esclavo',MASTER_LOG_FILE='mysql-bin.000002',  
MASTER_LOG_POS=980, MASTER_PORT=3306;
```

Si ejecutamos `SHOW SLAVE STATUS\G`, no obtendremos ningún error y por lo tanto podrá copiar los archivos, pudiendo adoptar cualquier papel (maestro o esclavo) en cualquier momento.



```
Seconds_Behind_Master: 0  
Master_SSL_Verify_Server_Cert: No  
Last_IO_Errno: 0  
Last_IO_Error:
```