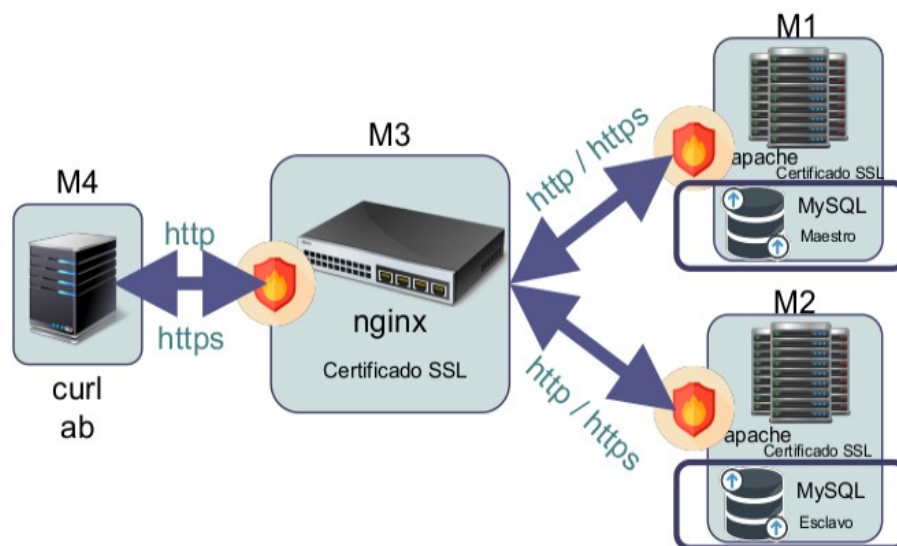




## SERVIDORES WEB DE ALTAS PRESTACIONES

### PRÁCTICA 5: REPLICACIÓN DE BASES DE DATOS MYSQL



David Armenteros Soto

# ÍNDICE

- 1. Introducción: Objetivos y cuestiones a resolver**
- 2. Crear BD e insertar datos con MySQL**
- 3. Replicar una BD MySQL con mysqldump**
- 4. Configuración Maestro-Esclavo**
- 5. Configuración Maestro-Maestro**
- 6. Configuración IPTABLES para puerto 3306**

## 1. Introducción: Objetivos y cuestiones a resolver

El objetivo principal de la práctica es la creación y gestión de una base de datos MySQL en las distintas máquinas servidoras. Realizaremos una copia de seguridad de nuestra base de datos replicando el contenido mediante una configuración maestro-esclavo y maestro-maestro.

Los objetivos concretos de esta práctica son:

- Crear una **base de datos** con al menos una tabla y algunos datos.
- Realizar una copia de seguridad de la BD completa usando **mysqldump** en la máquina principal y copiar el archivo de copia de seguridad a la máquina secundaria.
- **Restaurar** dicha copia de seguridad en la segunda máquina, de forma que en ambas máquinas esté esa BD de forma idéntica.
- Realizar la configuración **maestro-esclavo** de los servidores MySQL en M1 y M2 para que la replicación de datos se realice automáticamente.
- Añadir reglas **IPTABLES** para permitir tráfico al puerto **3306**
- Realizar la configuración **maestro-maestro** entre las dos máquinas de bases de datos.

## 2. Crear BD e insertar datos con MySQL

Antes de empezar, es importante mencionar que en todo momento, usaremos la interfaz de línea de comandos MySQL como **root**

```
root@m1-daarso98:~# sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.33-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

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>
```

Creemos la base de datos estudiante y la tabla datos con los comandos **create database** y **create table**

```
mysql> create database estudiante;
Query OK, 1 row affected (0.08 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 (2.01 sec)

mysql> show tables;
+-----+
| Tables_in_estudiante |
+-----+
| datos                 |
+-----+
1 row in set (0.00 sec)
```

Insertamos nuestros propios datos personales en la tabla con **insert** para comprobar el funcionamiento

```
mysql> insert into datos(nombre,apellidos,usuario,email) values ("David","Armenteros Soto","daarso98",
"daarso98@correo.ugr.es");
Query OK, 1 row affected (0.22 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos      | usuario | email                      |
+-----+-----+-----+-----+
| David  | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

Para ver los tipos de datos de nuestra tabla podemos utilizar **describe**

```
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 (0.02 sec)
```

## ■ Opciones avanzadas de MySQL

Actualizamos la tabla datos con el comando **update**

```
mysql> update datos set nombre='pepito' where usuario='daarso98';
Query OK, 1 row affected (0.16 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos      | usuario | email                      |
+-----+-----+-----+-----+
| pepito | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Borramos tupla de la tabla con el comando **delete**

```
mysql> delete from datos where nombre='pepito';
Query OK, 1 row affected (0.19 sec)

mysql> select * from datos;
Empty set (0.00 sec)
```

Borramos la tabla con el comando **drop**

```
mysql> Drop table datos;
Query OK, 0 rows affected (0.40 sec)

mysql> show tables;
Empty set (0.00 sec)
```

Creamos una segunda tabla llamada datos2 en la que establecemos una clave primaria con **PRIMARY KEY**

```
mysql> create table datos2 (nombre varchar(100), apellidos varchar(100), usuario varchar(100), email
varchar(100), PRIMARY KEY (usuario) );
Query OK, 0 rows affected (0.61 sec)

mysql> show tables;
+-----+
| Tables_in_estudiante |
+-----+
| datos                |
| datos2               |
+-----+
2 rows in set (0.00 sec)

mysql> describe datos2;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nombre     | varchar(100)  | YES  |     | NULL    |       |
| apellidos  | varchar(100)  | YES  |     | NULL    |       |
| usuario    | varchar(100)  | NO   | PRI | NULL    |       |
| email      | varchar(100)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

### 3. Replicar una BD MySQL con mysqldump

MySQL ofrece una herramienta para clonar las BD que tenemos en nuestra máquina. Esta herramienta es **mysqldump** y es parte de los programas del cliente MySQL que puede ser utilizado para generar copias de seguridad de BD.

Una cosa a tener en cuenta a la hora de realizar una copia de seguridad es que los datos pueden estar actualizándose constantemente en el servidor de la BD principal. En este caso, antes de hacer la copia de seguridad en el archivo **.sql** debemos evitar que se acceda a la BD para cambiar nada.

```
mysql> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0.06 sec)
```

Ahora ya si podemos ejecutar el comando **mysqldump** para guardar los datos

```
root@m1-daarso98:~# sudo mysqldump estudiante -u root -p > /tmp/estudiante.sql
Enter password:

--
-- Table structure for table `datos`
--
DROP TABLE IF EXISTS `datos`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `datos` (
  `nombre` varchar(100) DEFAULT NULL,
  `apellidos` varchar(100) DEFAULT NULL,
  `usuario` varchar(100) DEFAULT NULL,
  `email` varchar(100) DEFAULT NULL,
  ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `datos`
--
LOCK TABLES `datos` WRITE;
/*!40000 ALTER TABLE `datos` DISABLE KEYS */;
INSERT INTO `datos` VALUES ('David','Armenteros Soto','daarso98','daarso98@correo.ugr.es');
/*!40000 ALTER TABLE `datos` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2021-05-05  8:54:44
```

Una vez realizada la copia de seguridad, debemos desbloquear las tablas

```
mysql> UNLOCK TABLES;
Query OK, 0 rows affected (0.00 sec)

mysql> quit;
Bye
```

Copiamos el archivo **.sql** con todos los datos desde la máquina principal a la secundaria

```
root@m1-daarso98:/home/daarso98# sudo scp -P 2222 -i .ssh/mykey /tmp/estudiante.sql daarso98@192.168.1.56:105:/tmp/estudiante.sql
daarso98@192.168.1.56:105:/tmp/estudiante.sql
100% 1992  75.5KB/s  00:00
```

Con el archivo **.sql** en el esclavo ya podemos importar la BD completa en MySQL con los siguientes comandos

```
*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2021-05-05  8:59:40
root@m2-daarso98:/home/daarso98# sudo mysql -u root -p estudiante < /tmp/estudiante.sql
Enter password:
```

```
root@m2-daarso98:/home/daarso98# sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.33-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

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> 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> show tables;
+-----+
| Tables_in_estudiante |
+-----+
| datos                 |
+-----+
1 row in set (0.00 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos      | usuario | email                               |
+-----+-----+-----+-----+
| David  | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

## ■ Opciones avanzadas de mysqldump

Para copiar unas determinadas tablas de la base de datos, tenemos que indicarlo con la siguiente sentencia

```
root@m1-daarso98:/home/daarso98# sudo mysqldump estudiante -u root -p datos > /tmp/estudiante.sql
Enter password:
```

Para ignorar unas determinadas tablas de la base de datos con **--ignore-table**

```
root@m1-daarso98:/home/daarso98# sudo mysqldump estudiante -u root -p --ignore-table=estudiante.datos > /tmp/estudiante.sql
Enter password:
```

Para volcar la estructura de las tablas pero no sus datos con **--no-data**

```
root@m1-daarso98:/home/daarso98# sudo mysqldump estudiante -u root -p --no-data > /tmp/estudiante.sql
Enter password:
```

Para evitar el bloqueo de tablas con **--lock-tables=false**

```
root@m1-daarso98:/home/daarso98# sudo mysqldump estudiante -u root -p --lock-tables=false > /tmp/estudiante.sql
Enter password:
```

Para crear base de datos con datos binarios con **--hex-blob**

```
root@m1-daarso98:/home/daarso98# sudo mysqldump estudiante -u root -p --hex-blob > /tmp/estudiante.sql
Enter password:
```

## 4. Configuración Maestro-Esclavo

MySQL tiene la opción de configurar el demonio para hacer replicación de las BD sobre un esclavo a partir de los datos que almacena el maestro. Es un proceso automático muy adecuado para el entorno en el que nos encontramos. Para llevar a cabo la configuración **maestro-esclavo** las tareas a realizar son las siguientes:

- Como root editamos el archivo **/etc/mysql/mysql.conf.d/mysqld.cnf** de M1

```
GNU nano 2.9.3 /etc/mysql/mysql.conf.d/mysqld.cnf Modified
pid-file           = /var/run/mysqld/mysqld.pid
socket             = /var/run/mysqld/mysqld.sock
port              = 3306
basedir            = /usr
datadir            = /var/lib/mysql
tmpdir             = /tmp
lc-messages-dir    = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address       = 127.0.0.1
```



```
#log_error = /var/log/mysql/error.log
#
```

```
server-id          = 1
log_bin            = /var/log/mysql/mysql-bin.log
```

- Reiniciamos el servicio y comprobamos que no haya errores

```
root@m1-daarso98:/home/daarso98# sudo service mysql restart
root@m1-daarso98:/home/daarso98# sudo service mysql status
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2021-05-05 09:26:58 UTC; 8s ago
     Process: 31852 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pid (code=exi
     Process: 31822 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCE
   Main PID: 31854 (mysqld)
     Tasks: 27 (limit: 1107)
    CGroup: /system.slice/mysql.service
            └─31854 /usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pid

May 05 09:26:55 m1-daarso98 systemd[1]: Starting MySQL Community Server...
May 05 09:26:58 m1-daarso98 systemd[1]: Started MySQL Community Server.
```

- Como root editamos el archivo `/etc/mysql/mysql.conf.d/mysqld.cnf` de M2

```
GNU nano 2.9.3 /etc/mysql/mysql.conf.d/mysqld.cnf Modifie

# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

[mysqld_safe]
socket      = /var/run/mysqld/mysqld.sock
nice        = 0

[mysqld]
#
# * Basic Settings
#
user        = mysql
pid-file    = /var/run/mysqld/mysqld.pid
socket      = /var/run/mysqld/mysqld.sock
port        = 3306
basedir     = /usr
datadir     = /var/lib/mysql
tmpdir      = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address = 127.0.0.1
```

```
#log_error = /var/log/mysql/error.log
#
```

```
server-id          = 2
log_bin            = /var/log/mysql/mysql-bin.log
```

- Reiniciamos el servicio y comprobamos que no haya errores

```
root@m2-daarso98:/home/daarso98# sudo service mysql restart
root@m2-daarso98:/home/daarso98# sudo service mysql status
• mysql.service - MySQL Community Server
  Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2021-05-05 09:30:15 UTC; 18s ago
  Process: 24313 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pid (code=exit
  Process: 24276 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
  Main PID: 24315 (mysqld)
  Tasks: 27 (limit: 1107)
  CGroup: /system.slice/mysql.service
          └─24315 /usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pid

May 05 09:30:13 m2-daarso98 systemd[1]: Stopped MySQL Community Server.
May 05 09:30:13 m2-daarso98 systemd[1]: Starting MySQL Community Server...
May 05 09:30:15 m2-daarso98 systemd[1]: Started MySQL Community Server.
```

- Creamos el usuario **esclavo** en M1

```
mysql> CREATE USER esclavo_daarso98 identified by 'esclavo_daarso98';
Query OK, 0 rows affected (0.15 sec)

mysql> GRANT REPLICATION SLAVE ON *.* TO 'esclavo_daarso98'@'%' IDENTIFIED BY 'esclavo_daarso98';
Query OK, 0 rows affected, 1 warning (0.11 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.18 sec)

mysql> FLUSH TABLES;
Query OK, 0 rows affected (0.12 sec)

mysql> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0.00 sec)
```

- Mostramos la configuración del maestro y activamos las tablas de M1

```
mysql> SHOW MASTER STATUS;
+-----+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
+-----+-----+-----+-----+-----+
| mysql-bin.000001 |      998 |              |                  |                  |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- Configuramos el esclavo con los datos del maestro en M2

```
mysql> CHANGE MASTER TO MASTER_HOST='192.168.56.104',MASTER_USER='esclavo_daarso98',MASTER_PASSWORD='esclavo_daarso98',MASTER_LOG_FILE='mysql-bin.000001',MASTER_LOG_POS=998,MASTER_PORT=3306;
Query OK, 0 rows affected, 2 warnings (1.29 sec)
```

- Arrancamos el esclavo

```
mysql> START SLAVE;
Query OK, 0 rows affected (0.00 sec)
```

- Comprobamos el estado con **SHOW SLAVE STATUS\G** y comprobamos que el parámetro **Seconds\_Behind\_Master** sea 0

```

Master_SSL_Cipher:
Master_SSL_Key:
Seconds_Behind_Master: 0
Master_SSL_verify_server_cert: NO
Last_IO_Errno: 0
Last_IO_Error:
Last_SQL_Errno: 0
Last_SQL_Error:
Replicate_Ignore_Server_Ids:

```

Finalmente comprobamos el funcionamiento del este proceso, insertando una tupla en el **esclavo** y viendo como se muestra en el **maestro**

```

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> insert into datos(nombre,apellidos,usuario,email) values ("DavidPrueba","Prueba","daarso98","daar
daar
Query OK, 1 row affected (0.19 sec)

```

```

root@m2-daarso98:/home/daarso98# sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6
Server version: 5.7.33-0ubuntu0.18.04.1-log (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

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> 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 |
+-----+-----+-----+-----+
| David | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba | Prueba | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

## 5. Configuración Maestro-Maestro

A continuación, vamos a explicar como configurar de forma más **avanzada** el proceso de replicación de datos de la BD. En este caso, trataremos con la configuración **maestro-maestro** en la que se replican los datos de la BD sin importar la máquina servidora final que sea. De esta forma, conseguimos que M1 y M2 tengan la misma información en todo momento. Para ello, solo tenemos que realizar los pasos anteriormente realizados para la configuración maestro-esclavo pero en este caso el maestro será M1 y el esclavo M2 justo al contrario que antes. Con ello conseguimos que M1 y M2 sean maestros. Los pasos a seguir son los siguientes:

- Creamos el usuario **esclavo** en M2

```
root@m2-daarso98:/home/daarso98# sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.7.33-0ubuntu0.18.04.1-log (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

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 USER esclavo2_daarso98 IDENTIFIED by 'esclavo2_daarso98';
Query OK, 0 rows affected (0.13 sec)

mysql> GRANT REPLICATION SLAVE ON *.* TO 'esclavo2_daarso98'@'%' IDENTIFIED BY 'esclavo2_daarso98';
Query OK, 0 rows affected, 1 warning (0.14 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.14 sec)

mysql> FLUSH TABLES;
Query OK, 0 rows affected (0.10 sec)

mysql> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0.00 sec)
```

- Mostramos la configuración del maestro y activamos las tablas de M2

```
mysql> SHOW MASTER STATUS;
+-----+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
+-----+-----+-----+-----+-----+
| mysql-bin.000001 |      2009 |              |                  |                  |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- Configuramos el esclavo con los datos del maestro en M1

```
mysql> CHANGE MASTER TO MASTER_HOST='192.168.56.105',MASTER_USER='esclavo2_daarso98',MASTER_PASSWORD='esclavo2_daarso98',MASTER_LOG_FILE='mysql-bin.000001',MASTER_LOG_POS=2009,MASTER_PORT=3306;
Query OK, 0 rows affected, 2 warnings (0.51 sec)
```

- Arrancamos el esclavo

```
mysql> START SLAVE;
Query OK, 0 rows affected (0.08 sec)
```

- Comprobamos el estado con **SHOW SLAVE STATUS\G** y comprobamos que el parámetro **Seconds\_Behind\_Master** sea 0

```
Master_SSL_Cipher:
Master_SSL_Key:
Seconds_Behind_Master: 0
Master_SSL_verify_server_cert: NO
Last_IO_Errno: 0
Last_IO_Error:
Last_SQL_Errno: 0
Last_SQL_Error:
```

Finalmente, comprobamos el funcionamiento de la configuración insertando una tupla en un **maestro** y viendo como se replica en el otro **maestro**.

```
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> insert into datos(nombre,apellidos,usuario,email) values ("DavidPrueba2","PruebaMaestroMaestro",
"daarso98","daarso98@correo.ugr.es");
Query OK, 1 row affected (0.22 sec)

mysql> select * from datos;
```

nombre	apellidos	usuario	email
David	Armenteros Soto	daarso98	daarso98@correo.ugr.es
DavidPrueba	Prueba	daarso98	daarso98@correo.ugr.es
DavidPrueba2	PruebaMaestroMaestro	daarso98	daarso98@correo.ugr.es

```
3 rows in set (0.00 sec)
```

```
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                      |
+-----+-----+-----+-----+
| David       | Armenteros Soto | daarso98 | daarso98@correo.ugr.es    |
| DavidPrueba | Prueba         | daarso98 | daarso98@correo.ugr.es    |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre      | apellidos      | usuario  | email                      |
+-----+-----+-----+-----+
| David       | Armenteros Soto | daarso98 | daarso98@correo.ugr.es    |
| DavidPrueba | Prueba         | daarso98 | daarso98@correo.ugr.es    |
| DavidPrueba2 | PruebaMaestroMaestro | daarso98 | daarso98@correo.ugr.es    |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

## 6. Configuración IPTABLES para puerto 3306

Para realizar esta configuración, crearemos un script con las reglas **IPTABLES** en el que se permitirán conexiones de M1 y M2 habilitando la comunicación y flujo de información por el puerto 3306 que es el que usa MySQL para replicar sus BD.

```
GNU nano 2.9.3                               iptables3306.sh

#Eliminar todas las reglas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

#Política por defecto: permitir tráfico entrante
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -A INPUT -m state --state NEW,ESTABLISHED -j ACCEPT

#Permitir puerto 3306 para mysql
iptables -A INPUT -p tcp --dport 3306 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 3306 -j ACCEPT

iptables -L -n -v
```

Ejecutamos el script y comprobamos el funcionamiento

```
root@m2-daarso98:/home/daarso98# ./iptables3306.sh
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                 destination             state NEW,ESTABLISHED
 0      0 ACCEPT    all  --  *      *       0.0.0.0/0             0.0.0.0/0
Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                 destination
Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                 destination             tcp spt:3306
 0      0 ACCEPT    tcp  --  *      *       0.0.0.0/0             0.0.0.0/0
```

```
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 |
+-----+-----+-----+-----+
| David | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba | Prueba | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba2 | PruebaMaestroMaestro | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> insert into datos(nombre,apellidos,usuario,email) values ("DavidIPTABLES","PruebaIPTABLES","daarso98","daarso98@correo.ugr.es");
Query OK, 1 row affected (0.29 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos | usuario | email |
+-----+-----+-----+-----+
| David | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba | Prueba | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba2 | PruebaMaestroMaestro | daarso98 | daarso98@correo.ugr.es |
| DavidIPTABLES | PruebaIPTABLES | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos | usuario | email |
+-----+-----+-----+-----+
| David | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba | Prueba | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba2 | PruebaMaestroMaestro | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from datos;
+-----+-----+-----+-----+
| nombre | apellidos | usuario | email |
+-----+-----+-----+-----+
| David | Armenteros Soto | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba | Prueba | daarso98 | daarso98@correo.ugr.es |
| DavidPrueba2 | PruebaMaestroMaestro | daarso98 | daarso98@correo.ugr.es |
| DavidIPTABLES | PruebaIPTABLES | daarso98 | daarso98@correo.ugr.es |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```