JUNIO DE 2021

UGR CURSO 2020-2021
ELENA ORTIZ MORENO
CORREO: elena97om@correo.ugr.es

PRÁCTICA 5: REPLICACIÓN DE BASES DE DATOS MYSQL

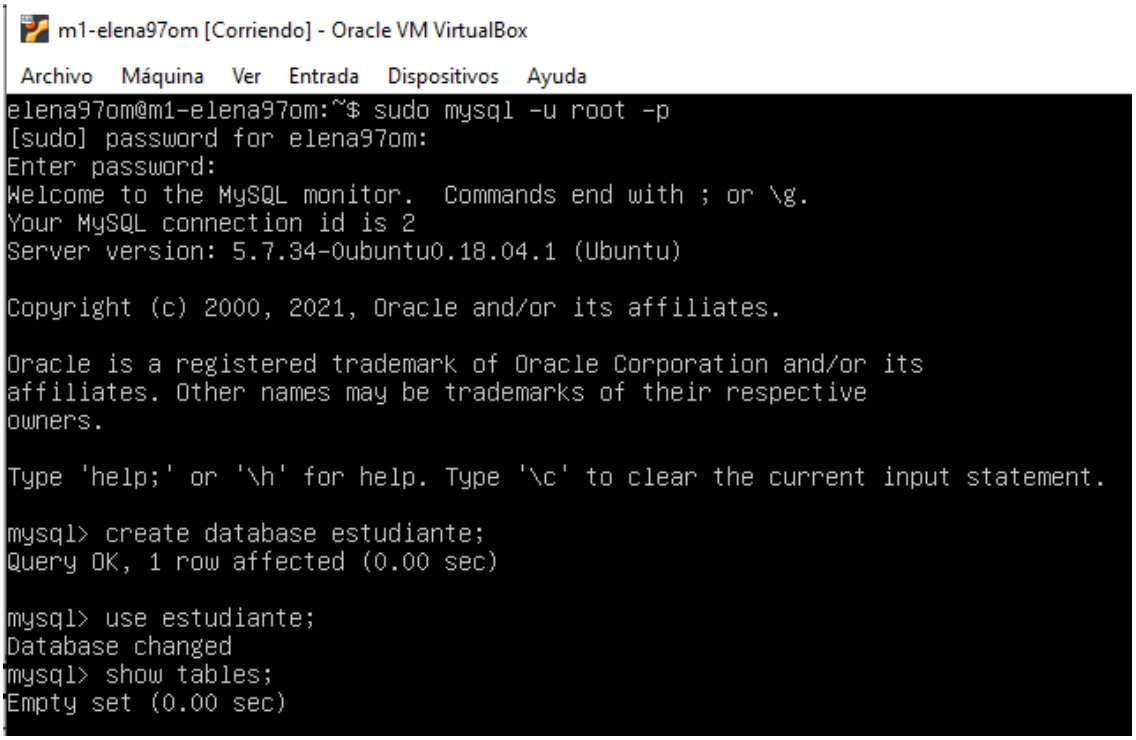
SERVIDORES WEB DE ALTAS PRESTACIONES

ÍNDICE:

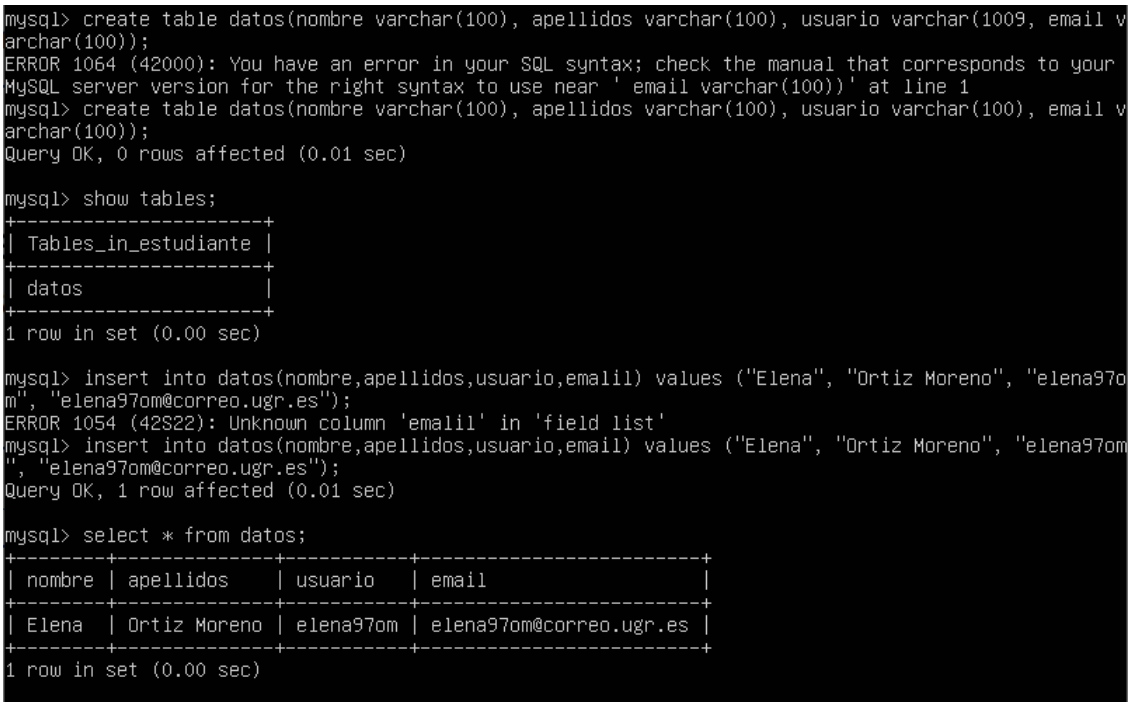
1. Crear una BD e insertar datos:

Vamos a crear una BD de ejemplo. Para ello usaremos, como root, la interfaz de líneas de comandos del MySQL.

1. Conectamos al servidor mysql, creamos BD estudiante y la seleccionamos.



2. Creamos la tabla datos e insertamos datos.



3. Ahora podemos ver los tipos de datos.

```
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)
```

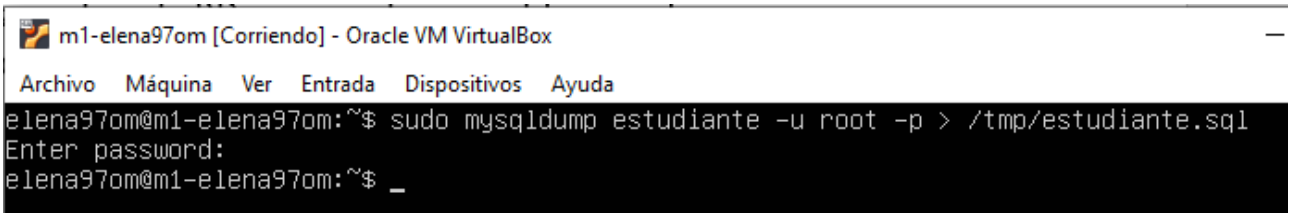
2. Replicar una BD MySQL con mysqldump

Mysqldump es una herramienda de MySQL para clonar bases de datos. Puede utilizarse para hacer copias de seguridad de las bases de datos.

Para replicar la BD primero hay que bloquearla para que no sea modificada mientras se realiza la copia de seguridad.

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

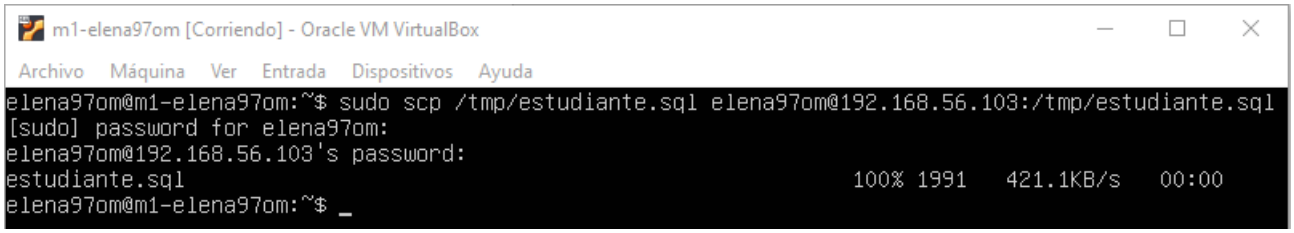
Ahora hacemos la copia de la BD al archivo .sql.



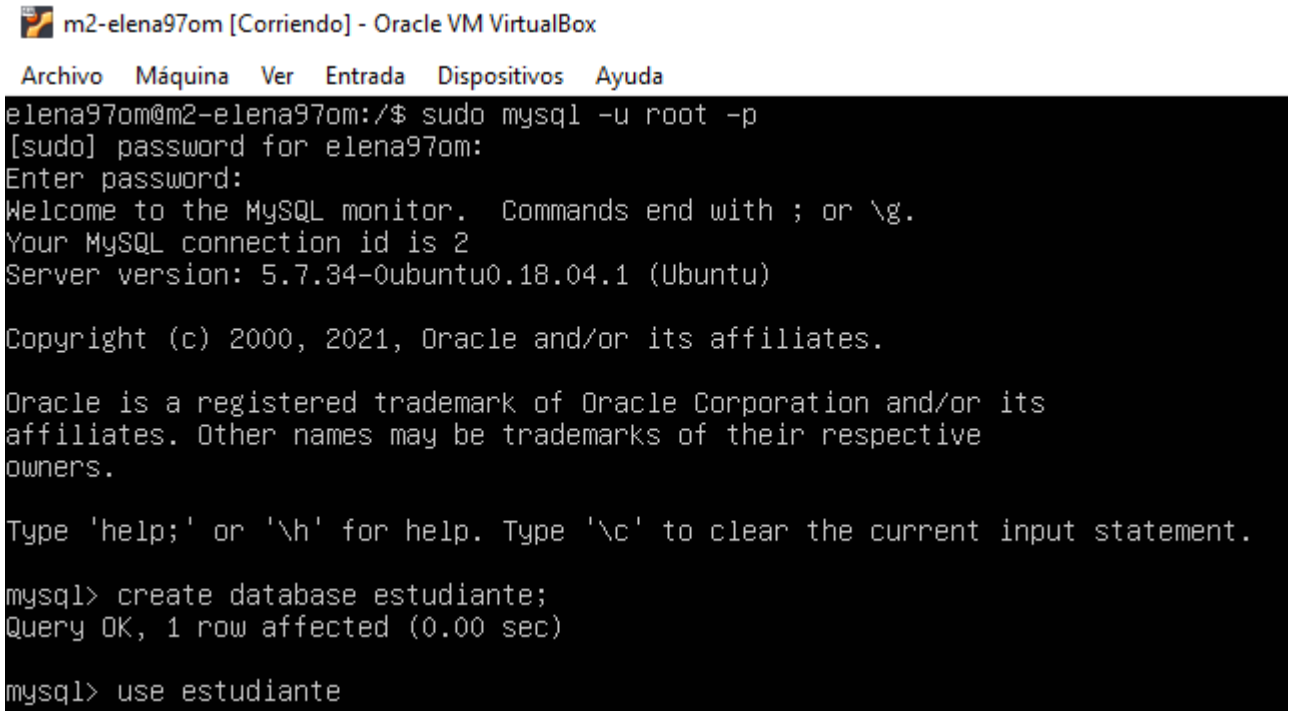
Y una vez hemos hecho la copia, desbloqueamos la BD.

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

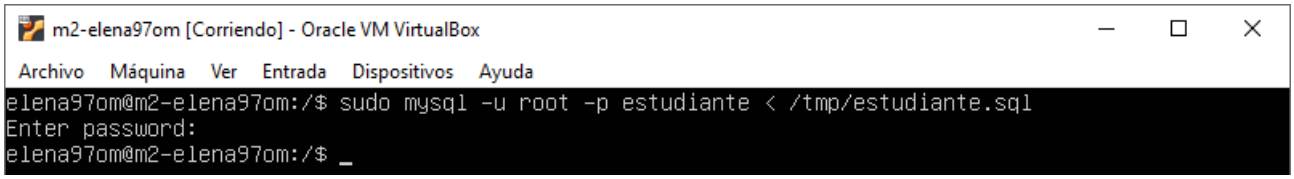
Una vez tenemos la copia de seguridad en el archivo .sql, tenemos que copiar este archivo en M2.



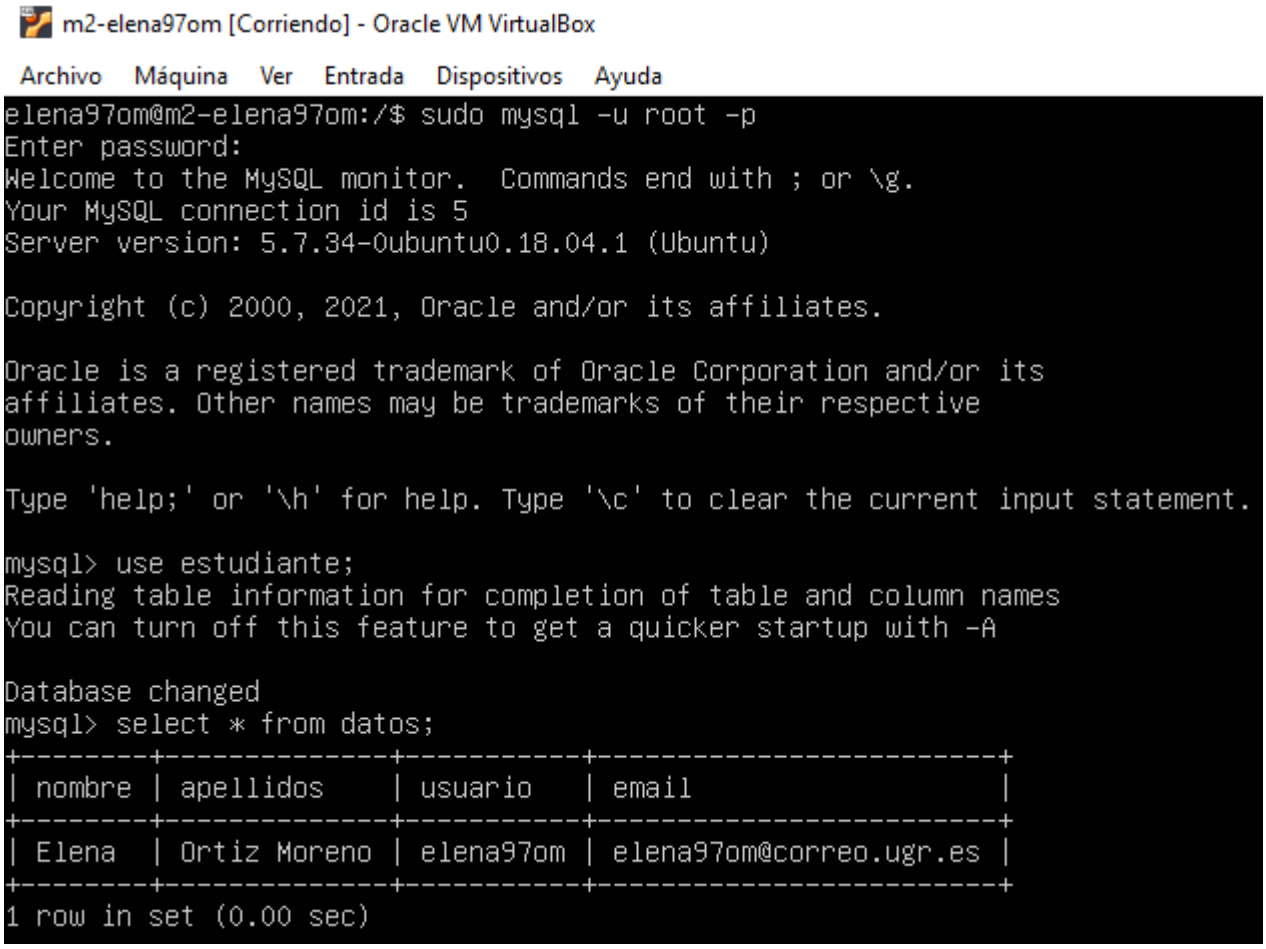
Ahora debemos crear en M2 la tabla estudiante para después restaurarla.



Y restauramos la base de datos.



Comprobamos que se haya realizado bien la restauración.

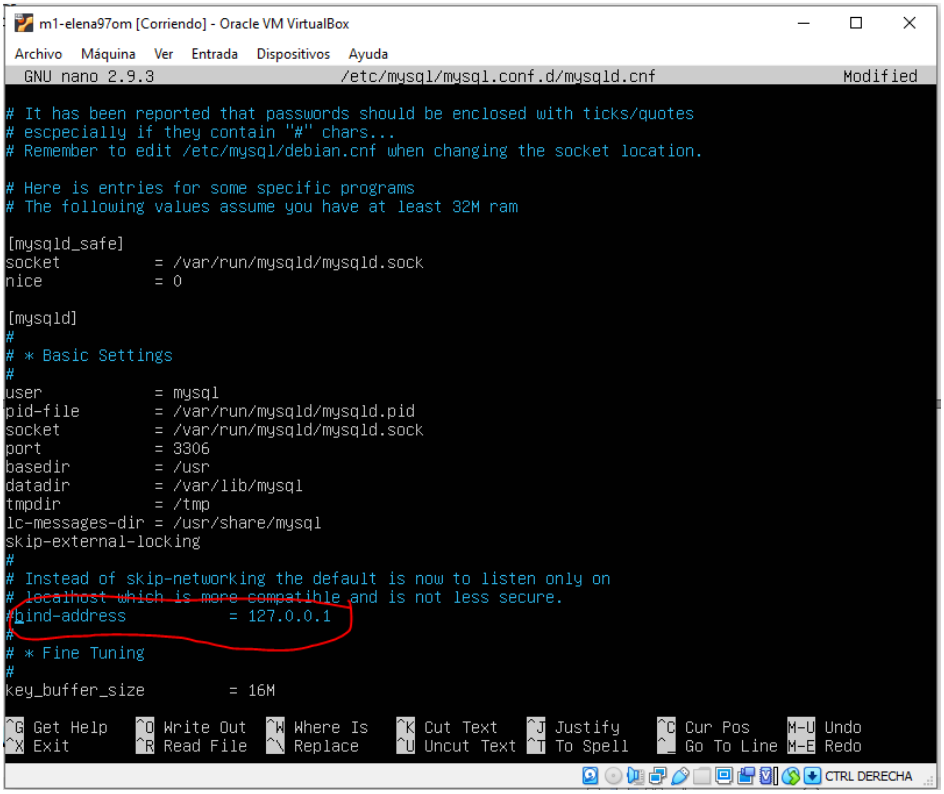


3. Replicar una BD mediante maestro-esclavo.

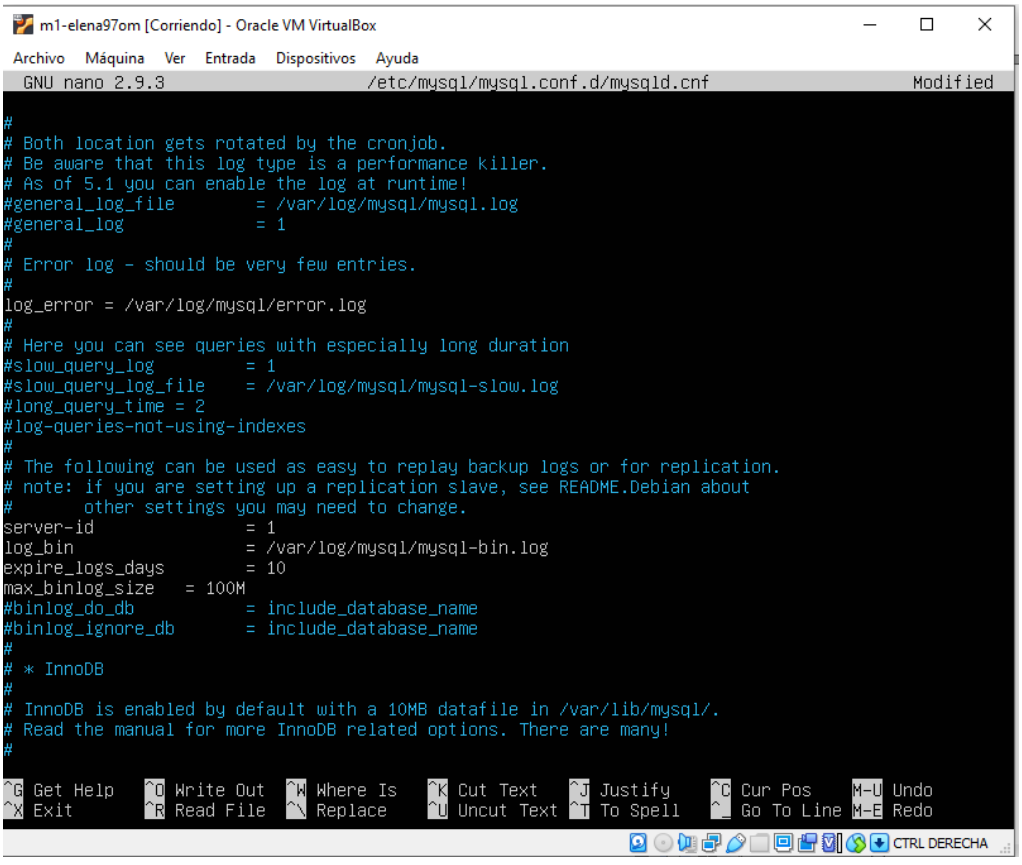
La opción anterior funciona, pero es algo que debe hacer un operador a mano. Para evitar esto, con MySQL se puede configurar un demonio para hacer la replicación de la BD sobre un esclavo a partir de los datos almacenados en el maestro. Para hacer esto, debemos configurar M1 como maestro y M2 como esclavo.

Vamos a empezar configurando M1 (maestro):

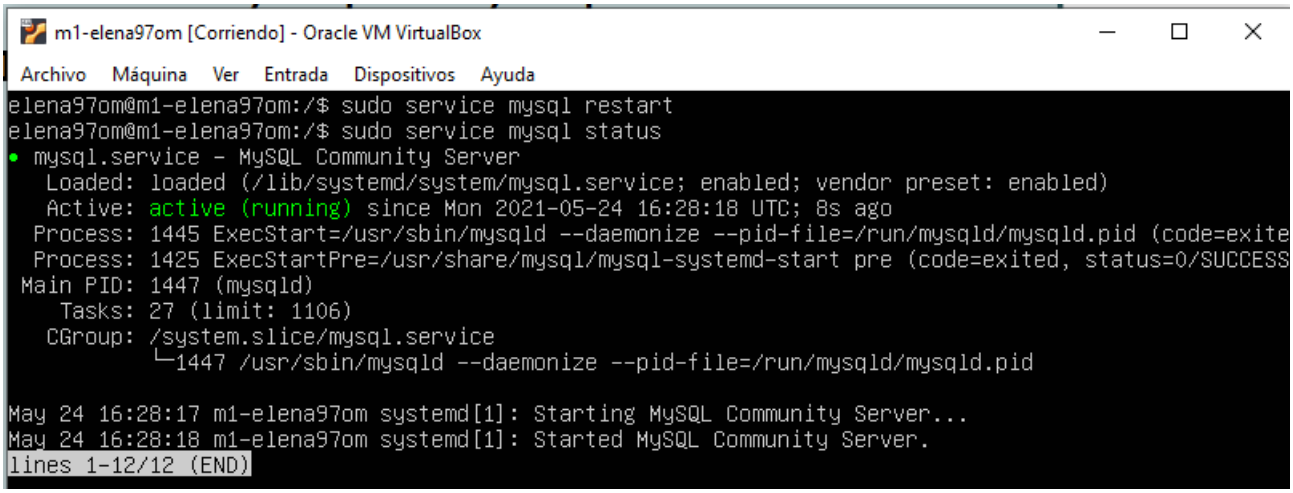
Primero comentamos el parámetro bind-address que sirve para que escuche un servidor.



Y después indicamos el archivo donde almacenar el log de errores, establecemos el indicador del servidor e indicamos el registro binario:

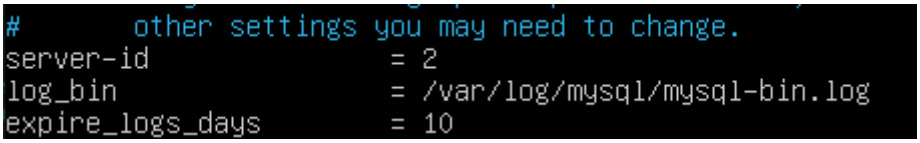


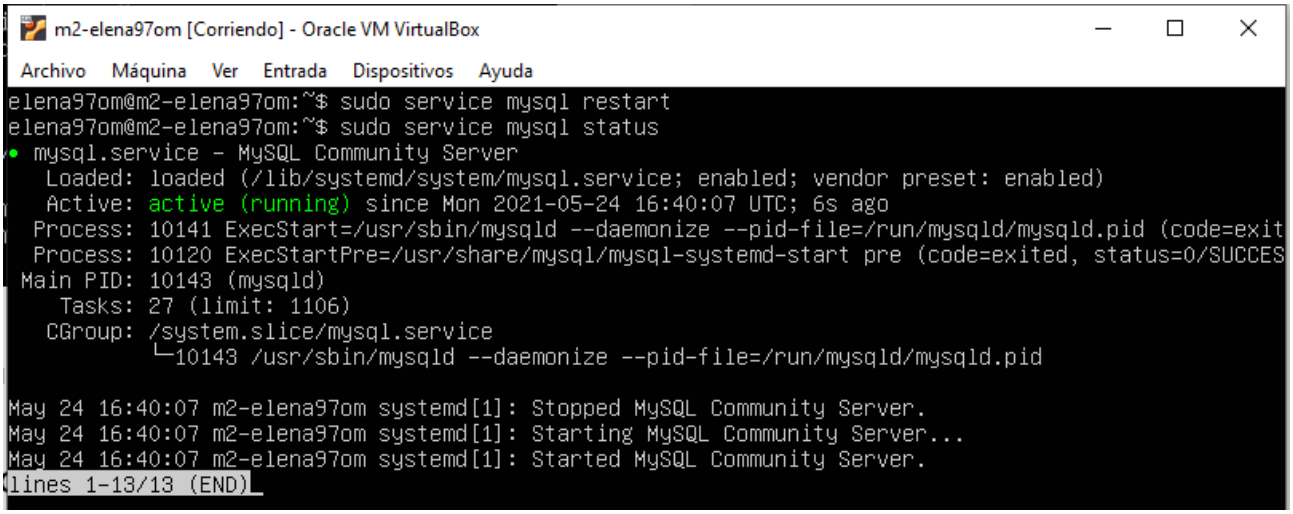
Ahora reiniciamos MySQL y comprobamos que no haya errores:



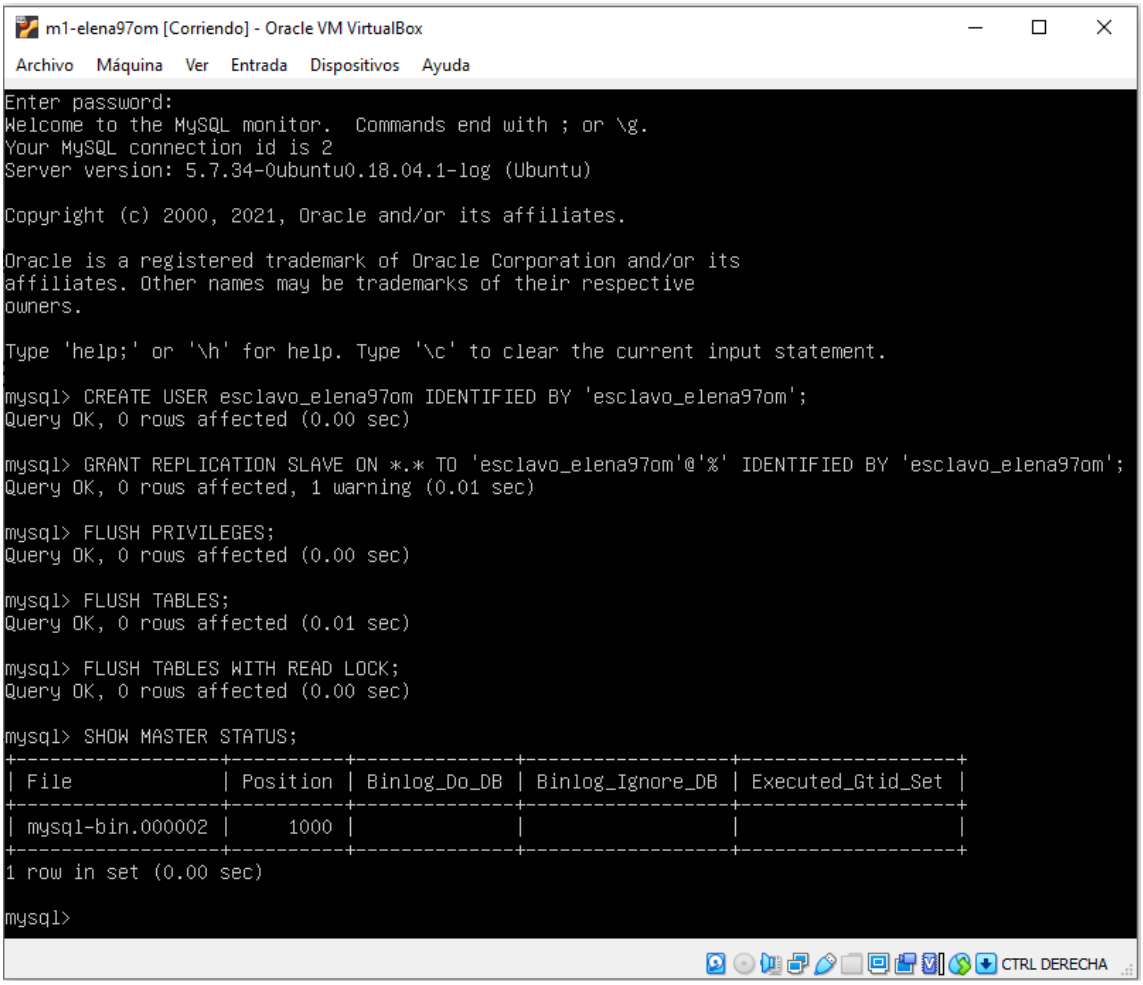
Una vez configurado el maestro, pasamos a configurar M2 (esclavo):

La configuración será similar, pero cambiando el identificador del servidor a 2 en vez de a 1.

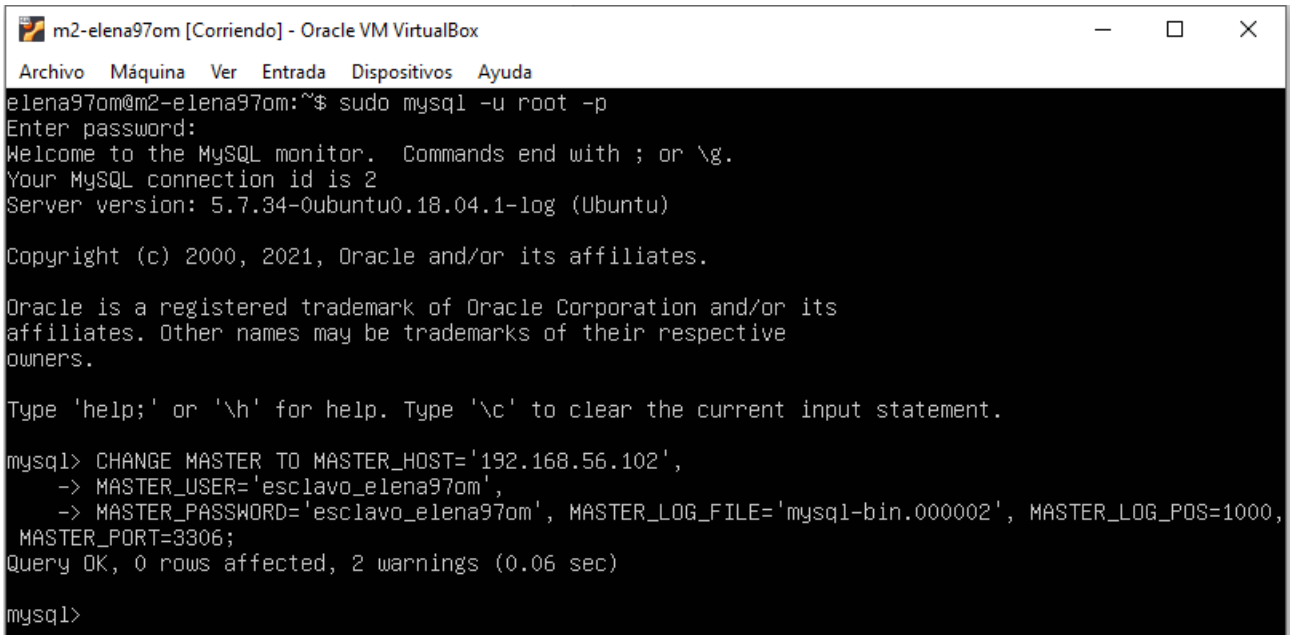




Volvemos a M1 para seguir configurando el maestro. Esta vez creamos un usuario “esclavo” para realizar la replicación:



Y volvemos a M2 para darle los datos del maestro.



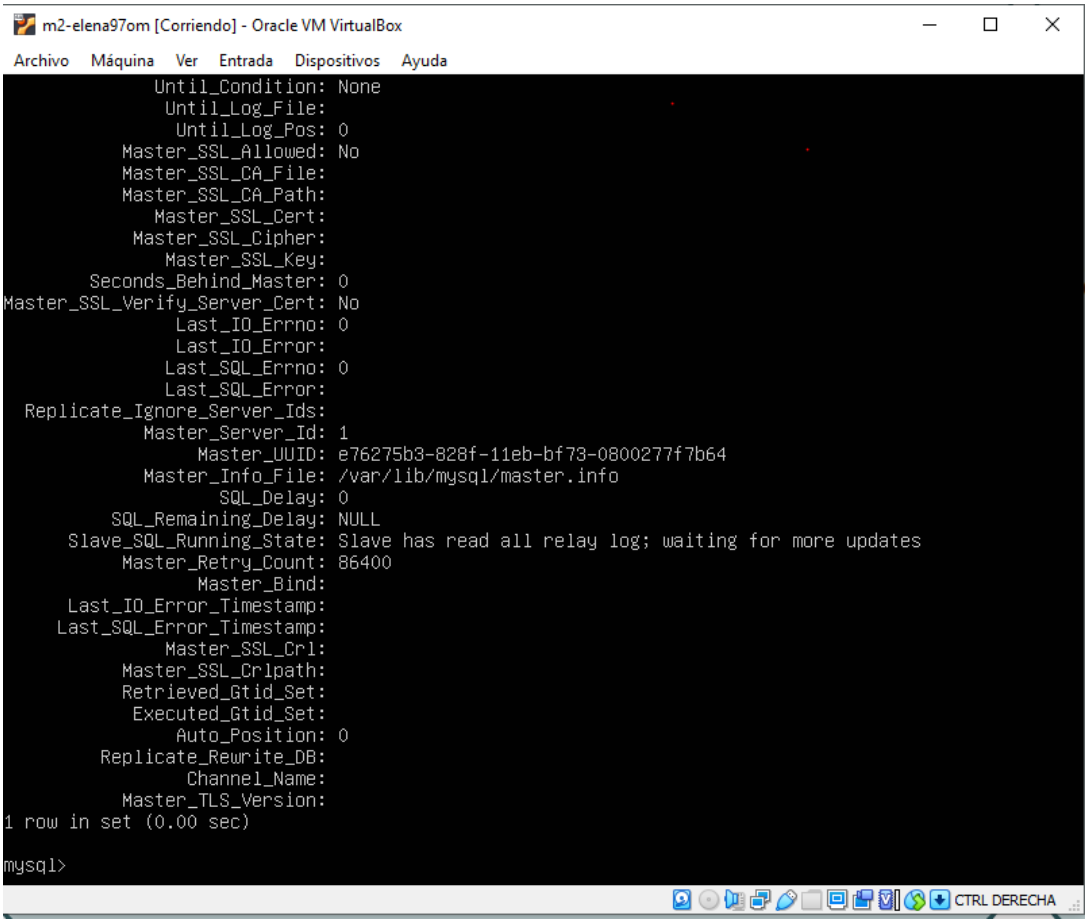
Y arrancamos el esclavo.

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

mysql> _
```

En M1 activamos las tablas de nuevo.

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

mysql>
```

Y en M2 comprobamos el estado del esclavo.



Como el valor de Seconds_Behind_Master es distinto de null y no me da ningún error, podemos ver que todo está funcionando perfectamente.