

# MySQL - MariaDb

Un mundo difícil, en el que las diferencias entre los motores SQL del tipo MySQL, comienzan a ser muy problemáticas. Incluso entre las versiones del mismo motor.

- [ERROR 1118 \(42000\) at line XXXXX: Row size too large \(> 8126\)](#)
- [Errores con MariaDB 10.3 al restaurar o hacer backups arrastrando versiones antiguas](#)
- [Como extraer de un backup de MySQL completo una base de datos y/o una tabla](#)
- [Como crear un usuario Mysql/MariaDB con Grant Privileges](#)
- [Restaurar mysqldump completo con problemas relacionados con VIEW y sus permisos](#)
- [Desactivación de las reglas de modos SQL de un servidor MySQL/MariaDB/Percona](#)

# ERROR 1118 (42000) at line XXXXX: Row size too large (> 8126)

## Introducción

Los upgrades de versión han sido un problema desde MySQL 5 y en MariaDB desde la 10. Muchas veces se quedan flecos que producen errores, que en caso de recuperación de desastres pueden ser un serio handicap.

## Error

```
[root@servidor02b mysql]# bunzip2 < dbdump.db.bz2 | mysql
ERROR 1118 (42000) at line 13300: Row size too large (> 8126). Changing some columns to TEXT or BLOB may
help. In current row format, BLOB prefix of 0 bytes is stored inline.
[root@servidor02b mysql]# bunzip2 < dbdump.db.bz2 | mysql
```

## Solución

Editamos el fichero de configuración de mysql (generalmente `/etc/my.cnf` o dentro de `/etc/mysql/`) para añadir en la sección `[mysqld]` el siguiente contenido (puedes adaptarlo a tu sistema, necesidades o posibilidades)

```
[mysqld]
innodb_log_file_size=512M
innodb_strict_mode=0
```

Después realizamos un restart del servidor e intentamos de nuevo el restore.

# Enlaces

- [Troubleshooting Row Size Too Large Errors with InnoDB](#)
- [Row size too large \(> 8126\). Changing some columns to TEXT or BLOB may help. In current row format, BLOB prefix of 0 bytes is stored inline](#)

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

# Errores con MariaDB 10.3 al restaurar o hacer backups arrastrando versiones antiguas

## Introducción

Los síntomas son variados, pero afectan todos a la base de datos `sys` la cual existía con anterioridad a la versión 10.3 de MariaDB y que ya está en desuso.

```
mysqldump: Got error: 1356: "View 'sys.host_summary' references invalid table(s) or column(s) or function(s) or definer/invoker of view lack rights to use them" when using LOCK TABLES
```

## Base de datos sys

Esta base de datos, no se incluye desde la 10.3, pero ha sido incorporada en la versión 10.6.0 (Alpha) y 10.6.1 (Beta), [bundle sys schema - MDEV-9077](#), y si no la usas (probablemente) es seguro eliminarla.

```
$ mysql
mysql > DROP DATABASE sys;
```

Si la necesitas, deberás intentar resolver los problemas descritos en los mensajes de error.

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de

[Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

# Como extraer de un backup de MySQL completo una base de datos y/o una tabla

## Introducción

Lo habitual en sistemas es tener un backup de bases de datos, por database o esquema. Pero a veces, sólo tenemos un backup completo. Mucho menos habitual es tener backups por tablas, así que en esta entrada, te explicamos cómo obtener el backup de una base de datos obtenida de un backup completo de mysqldump, y/o como extraer una tabla de un backup completo o de un esquema de la base de datos.

## Mysql --one-database

Una opción muy potente, que puede ser muy útil en ciertos escenarios (trabajo con logs binarios) pero que requiere un conocimiento profundo de MySQL. Así que mejor ir a lo práctico.

## Filtrar el fichero mysqldump con sed

Para mi es la mejor opción para extraer una base de datos o una tabla de un fichero SQL de mysqldump.

```
$ sed -n '/^-- Current Database: `nombre_de_la_base_de_datos`/,/^-- Current Database: `/p'
nombre_del_backup.sql > nombre_de_la_base_de_datos.sql
```

# Tip sed. Pasar una variable a un comando sed

El mismo comando se puede pasar de forma más fácil, usando variables. En mi caso uso este tip, aunque existen otras fórmulas para pasar variables, en este caso muchas de ellas fallaran por las comillas simples invertidas.

```
$ bd=nombre_de_la_bd
$ mydump=nombre_del_fichero_dump_sql
$ sed -n '/^-- Current Database: `"'${bd}`"/,/^-- Current Database: `/p' $mydump > ${bd}.sql
```

## Extraer una tabla de un fichero mysqldump

¿Necesitas restaurar una sólo tabla? Sencillo. Cambiamos un poco la estructura de la consulta (trabajando sobre el fichero de **una sola base de datos**)

```
$ tabla=nombre_de_la_tabla
$ sed -n '/^-- Table structure for table `"'${tabla}`"/,/^-- Table structure for table `/p' ${bd}.sql > ${tabla}.sql
```

## Nota de actualización para MacOS 2024/11/23

Algunas veces a sed en macos le da por mostrar problemas derivados la codificación. (Eso de no ser POSIX al MacOSX le mata a veces)

La solución (no lo intentes con las IA que a todas se les va la pinza...)

```
sed -i -n '/^-- Current Database: `"'${bd}`"/,/^-- Current Database: `/p' $mydump > ${bd}.sql
sed: RE error: illegal byte sequence
```

Solución

```
LC_ALL=C sed -i -n '/^-- Current Database: `"'${bd}`"/,/^-- Current Database: `/p' $mydump > ${bd}.sql
```

# Renombrar el backup de la tabla extraída para usarlo en la misma base de datos

Atención porque si queremos usar ese backup para crear una copia clonada de la tabla con otro nombre hay que modificar el fichero, ya que de lo contrario volcaremos el contenido en la misma tabla, y si es un backup antiguo, el lío esta servido.

```
-- Table structure for table `nombre_de_la_tabla`  
DROP TABLE IF EXISTS `nombre_de_la_tabla`;  
CREATE TABLE `nombre_de_la_tabla` (  
-- Dumping data for table `nombre_de_la_tabla`  
LOCK TABLES `nombre_de_la_tabla` WRITE;  
/*!40000 ALTER TABLE `nombre_de_la_tabla` DISABLE KEYS */;
```

Asi que debemos cambiarlo. Y una manía muy preocupante es la de usar editores de texto, sobre todo en windows, lo cual puede ser terrorífico, aparte de poco efectivo.

Usaremos **sed** otra vez.

```
# newtabla=nueva_tabla  
# sed -i -e 's/"${tabla}"/"${newtabla}"/g' ${tabla}.sql
```

## Enlaces

- [Passing a variable to sed](#)
- [How can I use variables in the LHS and RHS of a sed substitution?](#)

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de



[Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

# Como crear un usuario Mysql/MariaDB con Grant Privileges

## Introducción

La organización es muy adecuada en nuestro trabajo. Crear usuarios y bases de datos sin control alguno, y sin una nomenclatura es signo de desorganización, y el camino previo para las dificultades cuando necesitemos escalar nuestro proyecto. Además, a veces, no podemos dar permiso para todo (root) a todos. Eso sería un grave error en nuestra política de seguridad.

El siguiente artículo, ha sido creado con MariaDb 10.6 pero con algunas diferencias sutiles que podréis encontrar en la documentación de los respectivos motores y versiones.

Permite crear un usuario con privilegios suficientes para crear una base de datos y tener privilegios globales para es base de datos, siempre que cumpla, la nomenclatura de nombres basado en **prefijo\_**

## Crear una Database

Si no tienes una base de datos creada, tendrás que crearla.

Los comandos de este documento presuponen que tienes conocimientos básicos de MySQL.

```
# mysql
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 49
```

```
Server version: 10.6.3-MariaDB-1:10.6.3+maria~focal-log mariadb.org binary distribution
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

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

```
MariaDB [(none)]> CREATE DATABASE mibasedatos_pre;
```

Query OK, 1 row affected (0.000 sec)

```
MariaDB [(none)]> SHOW DATABASES;
```

```
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
| mibasedatos_pre  |
+-----+
6 rows in set (0.000 sec)
```

El comando accede como root porque lo tengo configurado a tal fin siendo root. Pude ser que en tu sistema tengas que usar `mysql -u root -p`

# Crear un usuario nuevo de MariaDB

```
MariaDB [(none)]> CREATE USER 'mibasedatos_usr'@'localhost' IDENTIFIED BY 'UnACoNtRaSeñaAdECuADa';
```

Query OK, 0 rows affected (0.001 sec)

Comprobamos

```
MariaDB [(none)]> SELECT User FROM mysql.user;
```

```
+-----+
| User      |
+-----+
| mariadb.sys |
| mysql      |
| root       |
```

```
| mibasedatos_pre |  
+-----+  
4 rows in set (0.001 sec)
```

# Otorgamos privilegios al user de MariaDB

```
MariaDB [(none)]> GRANT CREATE USER, CREATE ON *.* TO 'mibasedatos_usr'@'localhost' IDENTIFIED BY 'CKWor4Jh9CC4UskUg';  
MariaDB [(none)]> GRANT ALL PRIVILEGES ON `mibasedatos_%`.* TO 'mibasedatos_usr'@'localhost' WITH  
GRANT OPTION;  
Query OK, 0 rows affected (0.001 sec)
```

# Otorgamos privilegios al user en MySQL 8 +

```
CREATE USER, CREATE ON *.* TO 'mibasedatos_usr'@'localhost' IDENTIFIED BY 'CKWor4Jh9CC4UskUg';  
GRANT ALL PRIVILEGES ON *.* TO 'myroot'@'localhost';
```

# Comprobación

```
MariaDB [(none)]> USE mibasedatos_pre;  
Database changed  
MariaDB [mibasedatos_pre]> CREATE TABLE test (id INT);  
Query OK, 0 rows affected (0.003 sec)
```

Esto permite al usuario **crear cualquier base de datos** pero solo **usar aquellas que comienzan por el prefijo** Si deseas algo más específico en mi opinión, necesitas hacer un script que genere el usuario y los permisos sucesivos cuando cree una tabla, forzando al uso dle prefijo (estilo cpanel)

# Enlaces

- [How to Create MariaDB User and Grant Privileges](#)
- [MySQL granting privileges on wildcard database name to new user](#)

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

# Restaurar mysqldump completo con problemas relacionados con VIEW y sus permisos

## Mysqldump, restore, vistas y seguridad

Algunas veces, sobre todo cuando estamos trabajando con backups de otros servidores mysql, con problemas para restaurar un backup de una base de datos individual, que se hizo de forma completa.

Esta incluye las vistas, y por ende, los usuarios con permisos para las vistas, conocido en MySQL como `SQL SECURITY DEFINER` .

Si en nuestro sistema no existen esos usuarios, a los que hace referencia el fichero sql, obtendremos un fallo, en el volcado o restauración de la copia de seguridad, incluso si lo ejecutamos como root

Por ejemplo

```
ERROR 1356 (HY000) at line 1693: View 'database.view_condition_xx_table' references invalid table(s) or column(s) or function(s) or definer/invokee of view lack rights to use them
```

Podemos intentar algunos tips, que hay por ahí, incluso hacer un restore forzado pero nos quedaríamos con la duda de si está todo bien.

## Solución propuesta: desactivar al usuario afectado en su SQL SECURITY DEFINER

Primero para ver el tema y si este es el problema realmente (copiar y pegar artículos de internet no es buena idea si no se sabe lo que se hace) vamos a ver si los tiros van por ahí.

```
> cat my-database-dump.sql | grep -i DEFINER
/*!50013 DEFINER=`root`@`127.0.0.1` SQL SECURITY DEFINER */
/*!50013 DEFINER=`root`@`127.0.0.1` SQL SECURITY DEFINER */
/*!50013 DEFINER=`otrouser`@`%` SQL SECURITY DEFINER */
```

Vemos que existe un usuario que no existe en nuestro servidor, y ese es el portante el que produce el problema en el volcado.

La mejor opción en mi opinión, no es tratar de saltarse o modificar nada en el servidor, sino de obviar ese usuario, salvo que sea de interés crearlo en nuestro sistema, por otras causas.

## Desactivación del SQL SECURITY DEFINER problemático

Usando `sed` podremos cambiar el definer que nos falla a root y volver a intentar la restauración de la copia de seguridad de mysql que nos falló a causa `definer/invoker of view lack rights to use them`

```
sed -i.bak 's/DEFINER\=\`otrouser/DEFINER\=\`root/g' my-database-dump.sql
```

### Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido se entrega, tal y como está, sin que ello implique ninguna obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

# Desactivación de las reglas de modos SQL de un servidor MySQL/MariaDB/Percona

## Introducción

Hay una gran diferencia entre usar la desactivación de los modos SQL de un servidor SQL, por necesidad imperiosa, y otra muy distinta de persistir en la creencia de que es lo correcto.

Muchos de los modos SQL, cambia a lo largo de las vidas útiles de una versión de un sistemas de bases de datos, tipo SQL, y esto hace que en el momento más crucial ese cambio:

- Haga inservible, al menos rápidamente, un backup en un escenario de recuperación de desastres.
- No se pueda realizar una migración a un nuevo sistema por incompatibilidad de los datos contenidos.
- Por el fin de soporte de ese **workaround** que ha sido eliminado de la ecuación en la versión nueva.

## Consideraciones al Usar `sql_mode=""`:

1. Desactivación de Validaciones: Al establecer `sql_mode=""`, se desactivan validaciones como:

- `STRICT_TRANS_TABLES`: Permite que las inserciones y actualizaciones que no cumplen con los requisitos de tipo y tamaño se realicen, lo que potencialmente puede llevar a la corrupción de datos.
- `NO_ZERO_DATE`: Previene la inserción de fechas nulas o cero que suelen considerarse inválidas.



- **ONLY\_FULL\_GROUP\_BY:** Asegura que las consultas que usan GROUP BY cumplan con las reglas estándar SQL.

2. **Compatibilidad con PHP:** Si estás trabajando con PHP y experimentas problemas con la inserción de datos, a veces desactivar modos restrictivos puede solucionar problemas. Sin embargo, esto puede llevar a que se introduzcan datos erróneos o inconsistencia. Es importante identificar por qué los datos no se estaban insertando correctamente en primer lugar.
3. **Prácticas Recomendadas:** En lugar de desactivar todos los modos, es una buena práctica intentar configurar el sql\_mode de manera más específica. Por ejemplo, podrías excluir ciertos modos según sea necesario, en lugar de dejarlo vacío completamente:

```
sql_mode="STRICT_TRANS_TABLES,NO_ZERO_DATE"
```

## ## Configuración en my.cnf

Para configurar esto en el archivo `my.cnf`, puedes agregar o modificar la línea en la sección `[mysqld]`:

```
[mysqld]
sql_mode=""
```

# Reiniciar el Servidor

Después de realizar cambios en my.cnf, asegúrate de reiniciar el servidor de MySQL o MariaDB para que los cambios surtan efecto.

```
systemctl restart mariadb
```

# Comprobación

Acceso a MySQL shell.

```
MariaDB [(none)]> SHOW VARIABLES LIKE 'sql_mode';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| sql_mode      |      |
+-----+-----+
1 row in set (0.001 sec)
```

# Conclusión

Si decides establecer `sql_mode=""`, hazlo con precaución y asegúrate de revisar tu aplicación PHP para asegurarte de que todos los datos que se insertan sean válidos. También sería bueno investigar por qué ocurrían problemas sin establecer esto, ya que podrían haber soluciones más específicas y seguras sin necesidad de desactivar todas las validaciones.

## Links interesantes

- [Como ejecutar tu acceso a mysql en un Directadmin sin usar root](#)
- [Mysql conexión vía socket. Como saber donde está](#)

## Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).