

# QGIS, Navicat, y otros problemas de conexión con MySQL 8

## Introducción

Muchas veces no solemos encontrar respuesta a problemas de conexión con mysql con distintos programas, ya sea para acceso a mysql como herramientas de trabajo, on en mi caso para un programa como [QGIS](#) en el cual queriamos configurar una conexión mysql con un servidor basado en **MySQL 8**.

## Error

Error en QGIS MySQL

Críptico, inútil, y deficitario mensaje de error.

Pero es lo habitual.

Este me llamó la atención, ya que con otro servidor basado en Cpanel si conectaba. Así que revise la configuración MySQL.

```
mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2882
Server version: 8.0.30-0ubuntu0.20.04.2 (Ubuntu)

Copyright (c) 2000, 2022, 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> SHOW VARIABLES LIKE '%ssl%';
```

```
+-----+-----+
| Variable_name          | Value          |
+-----+-----+
| admin_ssl_ca           |                |
| admin_ssl_capath       |                |
| admin_ssl_cert         |                |
| admin_ssl_cipher       |                |
| admin_ssl_crl          |                |
| admin_ssl_crlpath      |                |
| admin_ssl_key          |                |
| have_openssl           | YES           |
| have_ssl               | YES           |
| mysqlx_ssl_ca          |                |
| mysqlx_ssl_capath      |                |
| mysqlx_ssl_cert        |                |
| mysqlx_ssl_cipher      |                |
| mysqlx_ssl_crl         |                |
| mysqlx_ssl_crlpath     |                |
| mysqlx_ssl_key         |                |
| performance_schema_show_processlist | OFF          |
| ssl_ca                 | ca.pem        |
| ssl_capath             |                |
| ssl_cert               | server-cert.pem |
| ssl_cipher             |                |
| ssl_crl                |                |
| ssl_crlpath            |                |
| ssl_fips_mode          | OFF           |
| ssl_key                | server-key.pem |
| ssl_session_cache_mode | ON            |
| ssl_session_cache_timeout | 300          |
+-----+-----+
27 rows in set (0.01 sec)
```

¡Qué bonito! La sutil diferencia estaba en que este servidor tiene activada la conexión SSL.

Es un servidor montado a pelo, sin panel de control, pero sin tunear la seguridad ya que como norma general, es sólo accesible a una serie de IP fijas y de hostnames. Por defecto está todo cerrado.

Como norma general animo a mis compañeros a usar un túnel SSH para conectar a mysql, pero aun así esto falla, y es por que a fin de cuentas con túnel o sin túnel la conexión trata de conectarse via SSL.

# Solución rápida: deshabilitar SSL en el servidor MySQL

Sólo tenemos que añadir `skip_ssl` a nuestro fichero de configuración del servidor y hacer un restart del servidor mysql

“ Esta es una solución rápida, bajo la premisa de que el tráfico entre puntos está bajo SSL (túnel) y que el acceso está reducido mediante políticas de acceso denegado-a-todos/abierto-sólo-autorizados

## Después de reiniciar

```
mysql> SHOW VARIABLES LIKE '%ssl%';
+-----+-----+
| Variable_name          | Value          |
+-----+-----+
| admin_ssl_ca           |                |
| admin_ssl_capath       |                |
| admin_ssl_cert         |                |
| admin_ssl_cipher       |                |
| admin_ssl_crl          |                |
| admin_ssl_crlpath      |                |
| admin_ssl_key          |                |
| have_openssl           | DISABLED      |
| have_ssl               | DISABLED      |
| mysqlx_ssl_ca          |                |
| mysqlx_ssl_capath      |                |
| mysqlx_ssl_cert        |                |
| mysqlx_ssl_cipher      |                |
| mysqlx_ssl_crl         |                |
| mysqlx_ssl_crlpath     |                |
| mysqlx_ssl_key         |                |
| performance_schema_show_processlist | OFF          |
| ssl_ca                 |                |
| ssl_capath             |                |
```

ssl_cert		
ssl_cipher		
ssl_crl		
ssl_crlpath		
ssl_fips_mode	OFF	
ssl_key		
ssl_session_cache_mode	ON	
ssl_session_cache_timeout	300	
+-----+-----+		

“ Este tip tiene una consecuencia. Si tenemos usuarios en nuestro servidor que usen Authentication **caching\_sha\_password** no podrán logearse con ese usuario, ya que el plugin requiere de un conexión segura. Si no podemos recrear el usuario con la autenticación estandar, no podrá logearse.

## Cambiar el modo de autenticación de un usuario

Si optamos por la via rápida, y ya tenemos algun uso del usuario en modo remoto con algun software (TablePlus, MySQLWorkbench, Navicat, DBeaver, ...) fallará. Si sigue siendo una via rápida valida, de acuerdo a nuestras configuraciones de seguridad, posibles usos, y otros elementos de valoración, podemos cambiar el usuario alterandolo en la linea de comandos de mysql.

```
mysql> ALTER user 'user'@'127.0.0.1' IDENTIFIED WITH mysql_native_password BY
'mypasswordcomplejo';
mysql > FLUSH PRIVILEGES;
```

## Solución: usar certificados SSL no autofirmados

- [Mysql 8, SSL con Let's Encrypt](#)
- Mysql 8, SSL con certificado de autoridad (Comprado) ToDo o por hacer...

### 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](#).

